Hello world program

- Write your first Cobol program. It only needs to print “Hello World”

Exercise 1

Your instructor should have created the following datasets:

- RSMnn.RSM.COBOL
- RSMnn.RSM.LOAD

In your library RSMnn.RSM.COBOL you will find the member #COMSTUD.

1. Edit this member replacing all occurrences of '??' with the suffix of your userid. e.g. C ?? 01 ALL.

2. Now replace all occurrences of PROGXX with PROG1.

   e.g. C PROGXX PROG1 ALL.

3. Submit your job and examine the Compilation job.

4. Now replace all occurrences of PROG1 with PROGXX to make it ready to use again later.

Exercise 2

Code the IDENTIFICATION DIVISION and ENVIRONMENT DIVISION for the program INREFORM.

The program is to read in a file containing student records (COURSIN) and write the same information, in a new format, to a second file (COURSOUT).

There is one output record written for every input record read.

This program should be stored in your COBOL library.

Exercise 3

Code the DATA DIVISION for the INREFORM program started earlier

Each file will have a fixed length of 80 bytes with the following layouts:

COURSE FILE (COURSIN)

COURSE-NAME	20 bytes, char
COURSE-DATE	8 bytes, numeric (DDMMYYYY)
STUDENT-NAME	18 bytes, char
FEE-PAID	6 bytes, numeric
unused	28 bytes, char

NEW COURSE FILE (COURSOUT)

STUDENT-NAME	18 bytes, char
COURSE-NAME	20 bytes, char
FEE-PAID	6 bytes, numeric
COURSE-DATE	8 bytes, numeric (DDMMYYYY)

Exercise 4

Complete the INREFORM program started earlier then compile, link edit and test it.

INREFORM should read in the file of student records described earlier (COURSIN) and output the same information to a second file (COURSOUT) in a slightly modified format. There should be one output record written for each input record read

Once you have coded and successfully compiled and linked the program, test it using the data found in member COURSIN in your RSMnn.RSM.COBOL dataset. The input data has the following format:

```
ADVANCED JCL         25012013FRED FLINTSTONE    000900
COBOL PROGRAMMING    01012013WILMA FLINTSTONE   003120
USING ISPF FOR FUN   13122012BARNEY RUBBLE      001500
BASIC BRAIN SURGERY  12042013BETTY RUBBLE       015000
INTRO. TO EMBALMING  31122012BAM BAM            123456
JCL FOR BEGINNERS    14012014DINO               002400
```

The output should be written to a new member (COURSOUT) in the same dataset. When your program has completed, examine the contents of this member. If you program has worked correctly the data should appear as follows:

```
FRED FLINTSTONE   ADVANCED JCL         00090025012013
WILMA FLINTSTONE  COBOL PROGRAMMING    00312001012013
BARNEY RUBBLE     USING ISPF FOR FUN   00150013122012
BETTY RUBBLE      BASIC BRAIN SURGERY  01500012042013
BAM BAM           INTRO. TO EMBALMING  12345631122012
DINO              JCL FOR BEGINNERS    00240014012014
```

Exercise 5

Write a new program called INPRINT, which is required to print out the Order Master File. The input file has the following format:

Order Master File:

Contents	Bytes	Data type
Record Identifier	1 - 12	character
Quantity Ordered	13 - 14	binary numeric - 0 decimal places
Stock Number	15 - 20	character
Unit Cost	21 - 24	packed-decimal - 2 decimal places
Percentage Discount	25 - 26	binary numeric - 2 decimal places
Gross Value	27 - 31	packed-decimal - 2 decimal places
Nett Value	32 - 36	packed-decimal - 2 decimal places
Delivery Date	37 - 44	numeric character (DDMMYYYY)

The report output should be in a format similar to that shown on the following page.

When you have completed the coding, compile, link edit and test the program. It is suggested that you write the program in 2 stages:

1. Initially do not attempt to solve the problem of headings, and multiple pages. Instead, simply produce a program that prints each record as a separate line on the report.

2. Once you have successfully produced the above report, amend your INPRINT program such that each page displays 50 records, with the headings shown on each page together with the current date and page number.

Exercise 5 - expected output

Expected output from program INPRINT:

```
RUN DATE 28/09/13                    ORDER REPORT                              PAGE 1
RECORD ID      QTY  NUMBER      COST    DISC       GROSS         NETT    DEL. DATE
------------  ----  ------  --------  -----  ----------  ----------  ----------
010111041301   200  123WHI     12.45   2.50     2490.00     2427.75  25/05/2012
010111041302    80  122GRE     36.00   2.50     2880.00     2808.00  25/05/2012
010111041303   125  113YEL     22.75   0.00     2843.75     2843.75  15/05/2012
010111041304    50  122WHI    120.00   4.50     6000.00     5730.00  03/05/2011
010111041305    50  111GRE     98.00   4.50     4900.00     4679.50  03/05/2011
010111041306     6  123BLU    175.00   5.00     1050.00      997.50  01/06/2011
010111041307     8  123YEL    133.33   7.50     1066.64      986.65  01/06/2012
010111052601   450  111WHI     15.00   0.00     6750.00     6750.00  01/06/2012
010111052602   600  113WHI      8.55   0.00     5130.00     5130.00  01/06/2012
010111052603   350  113BLU      6.67   0.00     2334.50     2334.50  01/06/2012
010111052604    25  123WHI     16.00   5.50      400.00      378.00  10/06/2012
011333040101    25  123WHI     16.00   5.50      400.00      378.00  01/06/2013
011777040102    50  221GRE     20.75   3.00     1037.50     1006.38  02/05/2013
Q11777040103   200  123WHI     12.45   2.50     2490.00     2427.75  25/05/2013
011777040104    80  122GRE     36.00   2.50     2880.00     2808.00  25/05/2013
011777040105   125  113YEL     22.75   0.00     2843.75     2843.75  15/05/2012
011777040106    50  122WHI    120.00   4.50     6000.00     5730.00  03/05/2012
011777040107    50  111GRE     98.00   4.50     4900.00     4679.50  03/05/2012
011777040108     6  123BLU    175.00   5.00     1050.00      997.50  01/06/2012
011777040109     8  123YEL    133.33   7.50     1066.64      986.65  01/06/2012
011777040110   450  111WHI     15.00  10.00     6750.00     6075.00  01/06/2012
011777040111   600  113WHI      8.55   0.00     5130.00     5130.00  01/06/2012
011777040112   350  113BLU      6.67   0.00     2334.50     2334.50  01/06/2012
011777040113    25  123WHI     16.00   5.50      400.00      378.00  01/06/2012
020222040901    50  221GRE     20.75   3.00     1037.50     1006.38  02/05/2012
020222041301   200  123WHI     12.45   2.50     2490.00     2427.75  25/05/2012
020222041302    80  122GRE     36.00   2.50     2880.00     2808.00  25/05/2012
020222041303   125  113YEL     22.75   0.00     2843.75     2843.75  15/05/2012
020222062701    50  122WHI    120.00   4.50     6000.00     5730.00  03/05/2012
020222062702    50  11GRE      98.00   4.50     4900.00     4679.50  03/05/2012
020222062703     6  123BLU    175.00   5.00     1050.00      997.50  01/06/2012
020222062704     8  123YEL    133.33   7.50     1066.64      986.65  01/06/2012
020222062901   450  111WHI     15.00   0.00     6750.00     6750.00  01/06/2012
020222062902   600  113WHI      8.55  10.00     5130.00     4617.00  01/06/2012
020222062903   350  113BLU      6.67   0.00     2334.50     2334.50  01/06/2012
020400440901    50  221GRE     20.75   3.00     1037.50     1006.38  02/05/2013
020400440902   200  123WHI     12.45   2.50     2490.00     2427.75  25/05/2013
020400440903   280  122GRE     36.00   2.50    10080.00     9828.00  25/05/2013
020400440904   125  113YEL     22.75   0.00     2843.75     2843.75  15/05/2013
020400440905    50  122WHI    120.00   4.50     6000.00     5730.00  03/05/2013
020400440906    50  111GRE     98.00   4.50     4900.00     4679.50  03/05/2012
020400440907     6  123BLU    175.00   5.00     1050.00      997.50  01/06/2012
020400440908     8  123YEL    133.33   7.50     1066.64      986.65  01/06/2012
020400470301   450  111WHI     15.00   0.00     6750.00     6750.00  01/06/2012
020400470302   600  113WHI      8.55  10.00     5130.00     4617.00  01/06/2012
020400470303   350  113BLU      6.67   0.00     2334.50     2334.50  01/06/2012
```

RECORD ID	QTY	NUMBER	COST	DISC	GROSS	NETT	DEL. DATE
020777041304	125	113YEL	22.75	0.00	2843.75	2843.75	15/05/2012
020777041305	50	122WHI	120.00	4.50	6000.00	5730.00	03/05/2012
020777041306	50	111GRE	96.00	4.50	4800.00	4584.00	03/05/2012
020777041307	6	123BLU	175.00	5.00	1050.00	997.50	01/06/2012
020777041308	8	123YEL	133.33	7.50	1066.64	986.65	01/06/2012
020777041309	450	111WHI	15.00	0.00	6750.00	6750.00	01/06/2012
020777041310	600	113WHI	8.55	0.00	5130.00	5130.00	01/06/2012
020777041311	350	113BLU	6.67	12.50	2334.50	2042.69	01/06/2012
020777041312	25	123WHI	16.00	5.50	400.00	378.00	01/06/2012
020777060901	50	221GRE	20.75	3.00	1037.50	1006.38	02/05/2012
020777060902	200	123WHI	12.45	2.50	2490.00	2427.75	25/05/2012
020777060903	80	122GRE	36.00	2.50	2880.00	2808.00	25/05/2012
030177031301	125	113YEL	22.75	0.00	2843.75	2843.75	15/05/2012
030177031302	50	122WHI	120.00	4.50	6000.00	5730.00	03/05/2012
030177031303	50	111GRE	98.00	4.50	4900.00	4679.50	03/05/2012
030177031304	26	123BLU	175.00	5.00	4550.00	4322.50	01/06/2012
030177031305	38	123YEL	133.33	7.50	5066.54	4686.55	01/06/2012
030177031306	450	111WEI	15.00	0.00	6750.00	6750.00	01/06/2012
030177031307	600	113WHI	8.55	0.00	5130.00	5130.00	01/06/2012
030177031308	350	113BLU	6.67	0.00	2334.50	2334.50	01/06/2012
030177031309	25	123WHI	16.00	5.50	400.00	378.00	01/06/2012
030177031310	50	221GRE	20.75	3.00	1037.50	1006.38	02/05/2012
030177041301	200	123WHI	12.45	2.50	2490.00	2427.75	25/05/2012
030177041302	60	122GRE	36.00	2.50	2160.00	2106.00	25/05/2012
030177041303	125	113YEL	22.75	0.00	2843.75	2843.75	15/05/2012
030177041304	50	122WHT	120.00	4.50	6000.00	5730.00	03/05/2012
030177041305	50	111GRE	98.00	4.50	4900.00	4679.50	03/05/2012
030177041306	6	123BLU	175.00	5.00	1050.00	997.50	01/06/2012
030177041307	8	123YEL	133.33	7.50	1066.64	986.65	01/06/2012
030977032601	450	111WHI	15.00	0.00	6750.00	6750.00	01/06/2010
030977032602	600	113WHI	8.55	0.00	5130.00	5130.00	01/06/2012
030977032003	350	113BLU	6.67	0.00	2334.50	2334.50	01/06/2012
030977032604	25	123WHI	16.00	5.50	400.00	378.00	11/06/2012
041777040901	50	221GRE	20.75	3.00	1037.50	1006.38	02/05/2013
041777040902	200	123WHI	12.45	2.50	2490.00	2427.75	25/05/2013
041777040903	80	122GRE	36.00	2.50	2880.00	2808.00	25/05/2013
041777040904	125	113YEL	22.75	0.00	2843.75	2843.75	15/05/2013
041777052701	50	122WHI	120.00	4.50	6000.00	5730.00	03/05/2012
041777052702	50	111GRE	98.00	4.50	4900.00	4679.50	03/05/2012
041777052703	6	123BLU	175.00	5.00	1050.00	997.50	01/06/2012
041777052704	8	123YEL	133.33	7.50	1066.64	986.65	01/06/2012
041777052705	450	111WHI	15.00	0.00	6750.00	6750.00	01/06/2012
041777062601	600	113WHI	8.55	10.00	5130.00	4617.00	01/06/2013
041777062602	35	113BLU	6.67	0.00	233.45	233.45	01/06/2012

COBOL programming case study - Part A

Background:

RSM Components Ltd. is a small company manufacturing electronic components. It has decided to computerise its payroll. You are to design and program the new system.

Part of the system is a series of 'management reports' from a sequential file that has a logical record length of 40 bytes.

The format of the file is:

Position	Contents	Characteristics	Notes
1 - 7	Employee No.	Numeric	(a)
8 - 20	Name	Alphabetic	
21	Initial 1	Alphabetic	
22	Initial 2	Alphabetic	
23	Sex	Alphabetic	
24 - 29	Date of birth	Numeric	(b)
30 - 33	Rate of pay	Numeric - 2 decimal places	
34 - 37	Hours worked	Numeric - 2 decimal places	(c)
38 - 40	Unused	Unused	

Notes:

a. The employee number is made up of three parts: Area (2 digits), Department (2 digits), clock-number (3 digits).
   Area numbers are 01-04.
   Department numbers are 01-05

b. The date of birth is in the form ddmmyy.

c. The hours worked are nominally for a 40 hour week though overtime and absence are be taken into account.

Case Study - Part A:

Write a program to produce a listing of staff at RSM Components. The report is to be presented in the format shown of the following page.

When you have completed your program your instructor will provide you with the name of a dataset containing values to test your program. It should give the results shown on the following page.

Case study part A - expected output

REPORT: CS1 RSM COMPONENTS LTD

EMP.NO.	NAME	INITS	SEX	D.O.B.
0101001	JONES	AA	M	010164
0101002	SMITH	AB	F	020265
0102001	JONSON	AC	F	030367
0102002	SMITHIES	AD	M	040451
0102003	JONES	AE	M	050552
0103001	SMITH	AF	F	060683
0103002	JONSON	AG	F	070794
0103003	SMITHIES	AH	M	080855
0104001	CLARK	AI	M	090986
0104002	TAYLOR	AJ	F	101057
0104003	DAVIS	AK	F	111168
0104004	DAVIES	AL	M	121279
0105001	WHITE	AA	M	010164
0105002	HENDRY	AB	F	020265
0105003	MUIR	AC	F	030367
0105004	BUCK	AD	M	040450
0201001	GOOCH	AA	M	010154
0201002	GOWER	AB	F	020255
0202001	BOTHAM	AC	F	030357
0202002	PRINGLES	AD	M	040451
0202003	FALDO	AE	M	050552
0203001	WOOSNAM	AF	F	060653
0203002	BECK	AG	F	070774
:	:	:	:	:
:	:	:	:	:
0403003	BARLOW	AG	F	070764
0404001	BALDWIN	AH	M	080875
0404002	JONES	AI	M	090976
0404003	SMITH	AJ	F	101077
0404004	JONSON	AK	F	111168
0405001	SMITHIES	AL	M	121269
0405002	AARON	AA	M	010260
0405003	BUNCE	AB	F	020260
0405004	CALLER	AC	F	030350
0405005	DAUNTSEY	AD	M	040486
0405006	BLACK	KJ	M	120370
0405007	BROWN	WW	F	231285
0405008	GREEN	AD	M	240156
0405009	GREY	R	M	210951

Exercise 6

Code, compile and test the program INNUMBER. The input data will contain four 10 byte numbers, which include decimal points, as follows:

```
0000050.050000010.010000030.030000005.50
```

The rest of the input record is blank.

You are required to write a program which, using a series of simple DISPLAY statements, will display:

- The total of the four numbers
- The largest number
- The smallest number
- The average of the number

Test the program using in-stream data with the values shown above.

The results should be as follows:

TOTAL : 0000095.59

BIGGEST : 0000050.05

SMALLEST: 0000005.50

AVERAGE : 0000023.90

COBOL programming case study - part B

Case Study - Part B

The weekly 'payroll' report is to be produced. This report is to show all employees with the hours they have worked, their hourly rate of pay, and their pay for the week.

Weekly payment is calculated by multiplying the hours worked by the hourly rate.

In addition, the average hours worked, average rate of pay and average salary per employee should also be calculated and displayed at the end of the report.

The report should be in the format of the example on the following page. The results from your program should also be the same as those shown, although once again some lines have been omitted for brevity.

Case study part B - expected output

REPORT: CS2 RSM COMPONENTS LTD

EMPLOYEE NUMBER	EMPLOYEE NAME	INITS	HOURS WORKED	HOURLY RATE	WEEKLY PAY
0101001	JONES	AA	40.00	19.55	782.00
0101002	SMITH	AB	50.00	18.44	922.00
0102001	JONSON	AC	45.45	19.00	863.55
0102002	SMITHIES	AD	28.45	10.00	284.50
0102003	JONES	AE	40.00	19.55	782.00
0103001	SMITH	AF	50.00	12.44	622.00
0103002	JONSON	AG	45.45	11.00	499.95
0103003	SMITHIES	AH	42.45	10.00	424.50
0104001	CLARK	AI	40.00	10.55	422.00
0104002	TAYLOR	AJ	50.00	19.44	972.00
0104003	DAVIS	AK	45.45	10.00	454.50
0104004	DAVIES	AL	48.45	14.00	678.30
0105001	WHITE	AA	40.00	15.55	622.00
:	:	:	:	:	
0403003	BARLOW	AG	45.45	15.00	681.75
0404001	BALDWIN	AH	42.45	14.00	594.30
0404002	JONES	AI	40.00	15.55	622.00
0404003	SMITH	AJ	50.00	14.44	722.00
0404004	JONSON	AK	45.45	15.00	681.75
0405001	SMITHIES	AL	48.45	14.00	678.30
0405002	AARON	AA	40.00	15.55	622.00
0405003	BUNCE	AB	50.00	14.44	722.00
0405004	CALLER	AC	45.45	15.00	681.75
0405005	DAUNTSEY	AD	28.45	14.00	398.30
0405006	BLACK	KJ	49.30	13.00	640.90
0405007	BROWN	WW	55.00	16.00	880.00
0405008	GREEN	AD	40.00	14.56	582.40
0405009	GREY	R	44.00	14.50	638.00
NUMBER OF EMPLOYEES 67		AVERAGE	44.00	14.62	646.56

Exercise 7

Amend the program INREFORM coded earlier showing the effect of an increase in course fees.

Courses which begin with the letters A to M inclusive will have their fees increased by 5%, while courses whose names begin with the letters N to Z will have their fees increased by 10%.

The program should produce a report of the following format::

```
ORIGINAL VALUES: FRED FLINTSTONE    ADVANCED JCL
000900 25012013
MODIFIED VALUES: FRED FLINTSTONE    ADVANCED JCL
000945 25012013

ORIGINAL VALUES: WILMA FLINTSTONE   COBOL PROGRAMMING
003120 01012013
MODIFIED VALUES: WILMA FLINTSTONE   COBOL PROGRAMMING
003276 01012013

ORIGINAL VALUES: BARNEY RUBBLE      USING ISPF FOR FUN
001500 13122012
MODIFIED VALUES: BARNEY RUBBLE      USING ISPF FOR FUN
001650 13122012

ORIGINAL VALUES: BETTY RUBBLE       BASIC BRAIN SURGERY
015000 12042013
MODIFIED VALUES: BETTY RUBBLE       BASIC BRAIN SURGERY
015750 12042013

ORIGINAL VALUES: BAM BAM            INTRO. TO EMBALMING
123456 31122012
MODIFIED VALUES: BAM BAM            INTRO. TO EMBALMING
129628 31122012

ORIGINAL VALUES: DINO               JCL FOR BEGINNERS
002400 14012014
MODIFIED VALUES: DINO               JCL FOR BEGINNERS
002520 14012014
```

Exercise 8

Code and test the program INVALDT. The purpose of this program is to read the incoming Transaction File (TRANFILE), inspect the individual fields and report any errors. Valid records will be written to an output file (VALRECS). Any record containing errors is to be included on the error report (ERRORS).

The format of the transaction file (TRANFILE) is as follows:

Position	Contents	Characteristics
1 - 20	RECORD-ID	The first character must be alphabetic.
21 - 28	DATE	Format DDMMYYYY.
		Check that DD is valid for month.
		Check that MM is in the range 1-12.
		No other validation is required.
29 - 38	NETT-VALUE	Must be a numeric value.
39 - 41	CODE	Must have one of the following values:
		AAA
		BBB
		CCC
		DDD
		WHI
		GRE
		BLU
		RED
		QWE
		RTY
		UIO
		ASD
42 - 45	SEQUENCE	Numeric. No validation required.

When you are ready to test your program your instructor will supply you with the name of a dataset containing the test data.

If your program works correctly, the output file (VALRECS) should contain the following two records:

```
      VALID INTEGER       0101199900000123400QWE0001
      VALID DECIMAL       1212199900000123456QWE0005
```

The error report (ERRORS) should contain the details shown on the following page.

```
RUN DATE 26/09/13                  ERROR REPORT
PAGE    1
<------------ INPUT RECORD ----------------->   <--- REASON
FOR ERROR --->
1NVALID RECID       01012013000001234QWE0002   INVALID
RECORD-ID
CORRUPT CODE        01012014000001234$$$0003   INVALID
CODE
NON NUMERIC         01012014AAAAAAAAAAQWE0004   NETT IS
NON-NUMERIC
SON OF CORRUPT CODE 12122013000123456BLA0006   INVALID
CODE
BAD DAY             31022013000001234QWE0007   INVALID
DATE - DAY
BAD MONTH           13132013000001234QWE0008   INVALID
DATE - MONTH
```

COBOL programming case study - part C

The ‘payroll’ run in Case study - Part B was incorrect. The following details had been forgotten:

- No overtime payments were included. The standard working week is 40 hours, and for any hours worked above this the overtime structure is:
  - 0 - 5 hours — base rate
  - 5 - 15 hours — base rate x 1.5
  - over 15 hours — base rate x 2

Recalculate the payroll and additionally format the output by area and department as shown on the example on the following page. As always, some lines have been omitted for brevity.

Case study part C - expected output

REPORT: CS3 RSM COMPONENTS LTD

AREA	DEPT	NUMBER	NAME	INITS	HOURS	RATE	PAYMENT
01	01	001	JONES	AA	40.00	19.55	782.00
01	01	002	SMITH	AB	50.00	18.44	968.10
			DEPARTMENT TOTAL				1750.10
01	02	001	JONSON	AC	45.45	19.00	867.82
01	02	002	SMITHIES	AD	28.45	10.00	284.50
01	02	003	JONES	AE	40.00	19.55	782.00
			DEPARTMENT TOTAL				1934.32
01	03	001	SMITH	AF	50.00	12.44	653.10
01	03	002	JONSON	AG	45.45	11.00	502.42
01	03	003	SMITHIES	AH	42.45	10.00	424.50
			DEPARTMENT TOTAL				1580.02
01	04	001	CLARK	AI	40.00	10.55	422.00
01	04	002	TAYLOR	AJ	50.00	19.44	1020.60
01	04	003	DAVIS	AK	45.45	10.00	456.75
01	04	004	DAVIES	AL	48.45	14.00	702.45
			DEPARTMENT TOTAL				2601.80
01	05	001	WHITE	AA	40.00	15.55	622.00
01	05	002	HENDRY	AB	50.00	18.44	968.10
01	05	003	MUIR	AC	45.45	19.00	867.82
01	05	004	BUCK	AD	28.45	10.00	284.50
			DEPARTMENT TOTAL				2742.42
			AREA TOTAL				10608.66
02	01	001	GOOCH	AA	40.00	10.55	422.00
02	01	002	GOWER	AB	50.00	10.44	548.10
			DEPARTMENT TOTAL				970.10
				:			
				:			
04	05	001	SMITHIES	AL	48.45	14.00	702.45
04	05	002	AARON	AA	40.00	15.55	622.00
04	05	003	BUNCE	AB	50.00	14.44	758.10
04	05	004	CALLER	AC	45.45	15.00	685.12
04	05	005	DAUNTSEY	AD	28.45	14.00	398.30
04	05	006	BLACK	KJ	49.30	13.00	668.85
04	05	007	BROWN	WW	55.00	16.00	960.00
04	05	008	GREEN	AD	40.00	14.56	582.40
04	05	009	GREY	R	44.00	14.50	638.00
			DEPARTMENT TOTAL				6015.22
			AREA TOTAL				12805.18

END OF REPORT

COBOL programming case study - part D

The annual review (of salaries) is to be calculated:

Those employees in	Area 01	3.5% increase
	Area 02	3.9% increase
	Area 03	4.2% increase
	Area 04	4.5% increase

Calculate the payroll and format the output as shown on the following report example where the 'Notional New Basic Salary' column is based on a standard forty-hour week with no overtime or any additional payments.

Case study part D - expected output

REPORT: CS4 RSM COMPONENTS LTD

AREA	NAME	INITS	RATE (OLD)	INCR. (%)	RATE (NEW)	NOTIONAL WEEKLY PAY
01	JONES	AA	19.55	3.50	20.23	809.20
01	SMITH	AB	18.44	3.50	19.08	763.20
01	JONSON	AC	19.00	3.50	19.66	786.40
01	SMITHIES	AD	10.00	3.50	10.35	414.00
01	JONES	AE	19.55	3.50	20.23	809.20
01	SMITH	AF	12.44	3.50	12.87	514.80
01	JONSON	AG	11.00	3.50	11.38	455.20
01	SMITHIES	AH	10.00	3.50	10.35	414.00
01	CLARK	AI	10.55	3.50	10.91	436.40
01	TAYLOR	AJ	19.44	3.50	20.12	804.80
01	DAVIS	AK	10.00	3.50	10.35	414.00
01	DAVIES	AL	14.00	3.50	14.49	579.60
01	WHITE	AA	15.55	3.50	16.09	643.60
01	HENDRY	AB	18.44	3.50	19.08	763.20
01	MUIR	AC	19.00	3.50	19.66	786.40
01	BUCK	AD	10.00	3.50	10.35	414.00
	AREA TOTAL					9808.00
			:			
			:			
04	JONES	AA	15.55	4.50	16.24	649.60
04	FOWLER	AB	14.44	4.50	15.08	603.20
04	BEALE	AC	15.00	4.50	15.67	626.80
04	PEARCE	AE	15.55	4.50	16.24	649.60
04	SUGDEN	AF	14.44	4.50	15.08	603.20
04	BARLOW	AG	15.00	4.50	15.67	626.80
04	BALDWIN	AH	14.00	4.50	14.63	585.20
04	JONES	AI	15.55	4.50	16.24	649.60
04	SMITH	AJ	14.44	4.50	15.08	603.20
04	JONSON	AK	15.00	4.50	15.67	626.80
04	SMITHIES	AL	14.00	4.50	14.63	585.20
04	AARON	AA	15.55	4.50	16.24	649.60
04	BUNCE	AB	14.44	4.50	15.08	603.20
04	CALLER	AC	15.00	4.50	15.67	626.80
04	DAUNTSEY	AD	14.00	4.50	14.63	585.20
04	BLACK	KJ	13.00	4.50	13.58	543.20
04	BROWN	WW	16.00	4.50	16.72	668.80
04	GREEN	AD	14.56	4.50	15.21	608.40
04	GREY	R	14.50	4.50	15.15	606.00
	AREA TOTAL					11700.40

END OF REPORT

COBOL programming case study - part E

If RSM Components reaches its annual profit target, each employee receives a bonus. This is calculated as two weeks basic salary, i.e. assuming that a 40 hour week has been worked; overtime or working weeks < 40 hours do not affect this calculation.

Additionally, each employee receives a small gift appropriate to their sex. The bonus and gift are given in the month of their birthday.

A report is to be produced showing the total cash amount and number of gifts required for each month of the year. An example of the required print layout and the expected results are shown on the following page:

Case study part E - expected output

REPORT: CS5 RSM COMPONENTS LTD

MONTH	CASH	GIFTS(F)	GIFTS(M)
JANUARY	9792.80	0	8
FEBRUARY	10805.60	8	1
MARCH	11280.00	8	1
APRIL	6080.00	0	6
MAY	5296.00	0	4
JUNE	4460.80	4	0
JULY	4480.00	4	0
AUGUST	5280.00	0	5
SEPTEMBER	5736.00	0	5
OCTOBER	5020.80	4	0
NOVEMBER	4400.00	4	0
DECEMBER	5760.00	1	4
TOTALS	78392.00	33	34

END OF REPORT