



(National Council for Vocational Awards)



Computer Programming C20013

Theory Examination 2002

Duration: Two Hours

NAME (PRINT): Washed Solution

EXAM NUMBER 12

INSTRUCTIONS TO CANDIDATES:

Answer all ten questions

Return this exam/answer paper when finished

Extra paper is available from the exam supervisor if required

This written exam counts as 50% of the total module

Answer all 10 questions. All questions carry equal marks.

1. This C program contains 4 errors that will stop it from compiling. Circle and/or list the errors.

① → Should be the same name

```

    3
    #include <stdio.h>
    int loopy;
    main ()
    {
        loopy = 1;
        printf ("+-----+\n");
        while (loopy <= 8)
        {
            printf ("|      | \n");
            loopy++;
        }
        printf ("+-----+\n");
    }
  
```

②

④

1	Declared variable name differs from name used in program
2	Inverted commas not closed
3	# include → ## missing
4	Should be loopy++; to increment the variable

2. Why is indentation used in programming?

To aid clarity of program; program is easier to read

What is the difference between an **int** and a **float** variable?

int is for whole numbers & float is for decimal values

3. The following C code will compile but will not generate the desired output. It has two errors. What are they? Circle and/or list the errors.

```
#include <stdio.h>
// This program should ask the users age, and inform
// them if they are 18 that they can now vote. This
// loop should execute 5 times.
int control, age;
float sum;
main ()
{
    for (control=1; control <= 5; control++);
    {
        printf ("Enter your age: ");
        scanf ("%d", &age);
        if (age = 18)
        {
            printf ("Now you can vote. Congratulations!\n");
        }
    }
}
```

² a loop never has a semicolon on the first line

1	Should be double(==) for comparison. Single = is for assignment.
2	Semicolon doesn't belong on first line of loop.

4. Assume that there are files and folders pre-existing on the linux system. Write the sequence of commands to be issued after telnetting on to the linux system to:

Task	Command/Key Sequence
Change directory to progfolder	cd progfolder
Delete the file assign04.bak	rm assign04.bak
Copy the file assign04.c to assign04.bak	cp assign04.c assign04.bak
Compile the file assign04.c	gcc assign04.c
Run the program	./a.out

5. The control variable for a **while** loop should appear in a program not less than four times. List those times:

- 1 declare
- 2 initialize
- 3 compare
- 4 program

6. Write the general form of the **if** statement:

```

if ( condition )
{
    action 1;    // condition is true
}
else
{
    action 2;    // condition is false
}

```

7. Indicate the values in each of the variables **a**, **b**, **c**, **d** and **e** after this program finishes:

```

#include <stdio.h>
main ()
{

```

start at 5 → $e = 5;$
 $d = 'A';$ → 65 = 'A' from ASCII table on page 8
 end at 2 → $while (e != 1)$
 {
 $a = e;$
 $b = (e * 2) + 1;$
 $e--;$
 }
 $c = a * b;$
 $d = d + b;$
 }
 $c = 2 * 5 = 10$
 $d = 65 + 5$
 ↑
 from $d = 'A';$

'a' doesn't change after this in loop
 $b = (5 * 2) + 1 = 11$ at start
 $b = (2 * 2) + 1 = 5$ at last loop

Variable	Value
a	2
b	5
c	10
d	70
e	1

loop program won't run again when 'e' gets to 1 - but it does get to 1

8. What output will the following program generate on screen?

```
#include <stdio.h>
int loop;
char thesymbol;
main ()
{
    thesymbol = 58;
    loop = 1;
    while (loop <= 5)
    {
        printf ("%c", thesymbol);
        thesymbol = thesymbol - 13;
        printf ("%c", thesymbol);
        thesymbol = thesymbol - 4;
        printf ("%c", thesymbol);
        printf ("\n");
        thesymbol = thesymbol + 17;
        loop++;
    }
}
```

use the ASCII
chart!

Write the output here:

: -)

: -)

: -)

: -)

: -)

5 smiley faces!

9. Write a C program snippet to read in a numeric value and a letter. The numeric value represents a distance. The letter entered will be either 'M' or 'K' – meaning that the distance value entered is in Miles or Kilometres. The letter entered may be in upper or lowercase.

Input letter	Conversion to perform
'M' or 'm'	Convert from Miles to Kilometres by dividing by 5 and multiplying by 8
'K' or 'k'	Convert from Kilometres to Miles by dividing by 8 and multiplying by 5

Display the converted value without any decimal places.

```
scanf ("%d %c", &dist, &unit);
if ( (unit == 'M') || (unit == 'm') )
{
    answer = (dist / 5) * 8;
}
else
{
    answer = (dist / 8) * 5;
}
printf ("Conversion gives: %d\n", answer);
```


10. Given that the formula to convert degrees Fahrenheit to degrees Celsius is given as:

must be complete

$$C = \frac{5 * (F - 32)}{9}$$

write a program which accepts a numeric value representing a temperature in degrees Fahrenheit, converts it to degrees Celsius using the formula above and writes out the converted value, with 2 places of decimals.

$$C = \frac{5 * (F - 32)}{9}$$

```
#include <stdio.h>
printf main (1
{
    float C, F;
    printf ("value? %f", &F);
    C = (5 * (F - 32)) / 9;
    printf ("Answer = %1.2f\n", C);
}
```


Figure 1. The ASCII table.

				032	SP	033	!	034	"	035	#
036	\$	037	%	038	&	039	'	040	(041)
042	*	043	+	044	,	045	-	046	.	047	/
048	0	049	1	050	2	051	3	052	4	053	5
054	6	055	7	056	8	057	9	058	:	059	;
060	<	061	=	062	>	063	?	064	@	065	A
066	B	067	C	068	D	069	E	070	F	071	G
072	H	073	I	074	J	075	K	076	L	077	M
078	N	079	O	080	P	081	Q	082	R	083	S
084	T	085	U	086	V	087	W	088	X	089	Y
090	Z	091	[092	\	093]	094	^	095	_
096	`	097	a	098	b	099	c	100	d	101	e
102	f	103	g	104	h	105	i	106	j	107	k
108	l	109	m	110	n	111	o	112	p	113	q
114	r	115	s	116	t	117	u	118	v	119	w
120	x	121	y	122	z	123	{	124		125	}
126	~	127	•								

Printable alphanumeric and punctuation characters used in normal document text

$58 = :$
 $58 - 13 = 45 = -$
 $45 - 4 = 41 =)$
 $41 + 17 = 58$ (back to start in question 8)