FETAC

Further Education and Training Awards Council

Comhairle na nDámhachtainí Breisoideachais agus Oiliúna

Comhairle na nDámhachtainí Breisoideachais agus Oiliúna

**Further Education and Training Awards Council**

# Computer Programming C20013

## April 2012

# Duration: Two Hours

**INSTRUCTIONS TO CANDIDATES:**

*Answer any **three** questions*

*All questions carry equal marks*

The use of calculators is not permitted

Answer the questions using the spaces provided in this booklet

Return this examination paper when finished

**This written exam counts as 40% of the total module**

NAME (PRINT): Suggested solutions

PPS NUMBER:____________________

DATE:____________________

## *Question 1. Total 40 marks.*

(a) This program contains 6 errors that will stop it from compiling. List the errors.

***6 * 5 marks***

```
1:    #!/usr/bin/perl
2:    use strict;
3:    #
4:    my (@words, $oneword, $count, $trap)
5:    @words=split(" ", "Fee Fi Foh Fum");
6:    $count = 0;
7:    foreach $oneword (@wordslist) {
8:      $count+;
9:      Print "$count $oneword \n"";
10:
11:    }
12:   }
```

| | |
|---|---|
| **1** | Line 4: missing ; at EOL |
| **2** | Line 7: variable @wordslist is not declared |
| **3** | Line 8: should be ++ |
| **4** | Line 9: should be small 'p' in print |
| **5** | Line 9: should be only 1 " at end of string being printed |
| **6** | Line 12: extra block closure bracket } |

(b) There is one error in this code snippet. Mark the error and identify it as either a *syntax* error or a *semantic* error. ***5 marks***

```
      print "Enter a number: ";
      $data = <STDIN>;
      if ($data = 5) { # Semantic error -> this should be ==
        print "SQuare is 25"; }
      else {
        print "Not matched"; }
```

(c) What is a reserved word? ***5 marks***

A reserved word is one which has a predefined meaning – make that word unusable as an identifier for any other purpose.

## *Question 2. Total 40 marks.*

(a) Write a **while** loop that writes out the even numbers from 20 to 100: ***10 marks***

```
$sentinel = 100;
$control = 20;
$step = 2;
while ($control <= $sentinel) {
        print “$control ”;
        $control++;
}
```

(b) Write the general form of the **if..else** statement: ***10 marks***

```
if (condition) {
        action if true;
}
else {
        action if false;
}
```

(c) The following perl code will compile and run but for any of at least 4 reasons will not generate the desired output. Why?

***4 * 5 marks***

```
#!/usr/bin/perl
# Convert sample sentence to all lowercase,
#
my ($sentence, $result, @letters, $letter);
$sentence = "THIS iS Some SAmPle TeXT.";
$result = "";
@letters = split ("", $sentance);
foreach $letter (@sentence) {
  if (($letter gt "A") && ($letter le "Z")) {
    $letter = chr(ord($letter)+65);
  }
  $result = $result.$letter;
}
print "$result\n";
```

| |
|---|
| 1 Scalar **$sentance** should be **$sentence** – strict substitutions would catch this |
| 2 Array **@letters** contains the individual letters – not **@sentence** |
| 3 Comparison operator **gt** should be **ge** |
| 4 Value **65** should be **32** |

## *Question 3. Total 40 marks.*

(a) What screen output is generated by this short program: ***10 marks***

```
#!/usr/bin/perl
printf "%c%c%c%c\n%c%c%c%c\n%c%c%c%c\n%c%c%c%c\n",
84,84,84,84,76,32,32,82,76,32,32,82,66,66,66,66;
```

```
TTTT
L  R
L  R
BBBB
```

(b) Indicate the values in each of the variables **$vr**, **$vs** and **$vt** after this program finishes:

***3 * 10 marks***

```
#!/usr/bin/perl
use strict;
my @nums=(4, 6, 8);
my ($vs, $vr, $vt);
my $index = 0;
$vr = $nums[0] * 1;
$vs = $nums[1] * 2;
$vt = $nums[2] * 3;
$vr = $index * 10;
$vs++;
while ($index <= 6)
{
  if ($index >= $nums[1]) {
    $vt=20;
  }
  $nums[2] = $index * 2;
  $index++;
}
$vs = $nums[0] * $index;
print "$vr, $vs, $vt\n";
```

| *Variable* | *Value* |
|---|---|
| **$vr** | 0 |
| **$vs** | 28 |
| **$vt** | 20 |

## *Question 4. Total 40 marks.*

| | |
|---|---|
| To convert degrees *Celcius* to degrees *Farenheit:*<br>*multiply* by 9, *divide* by 5, *add* 32 | F=(C*9)/5+32 |
| To convert degrees *Farenheit* to degrees *Celcius:*<br>*subtract* 32, *multiply* by 5, *divide* by 9 | C=(F-32)*5/9 |

Write a short perl program to:

1) Present a simple menu to show conversion options.

2) Take all steps to perform the conversion requested.

Include some error checking. Indent and comment as appropriate. ***40 marks***

*Figure 1. The ASCII table.*

| | | | | | |
|---|---|---|---|---|---|
| | | 032 SP | 033 ! | 034 " | 035 # |
| 036 $ | 37.00% | 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 | | | | | |