MA U11601 Quiz 01 8/10/21 ANSWERS

Answer any 3 questions. Submit them through Blackboard as pdfs, either handwritten and scanned, or typeset. They should be submitted before midnight on Monday 11 October. If more than three answers are submitted, only the first three will be marked. All questions carry 20 marks.

Show all work. That is, where an answer requires some calculation, show the calculation.

Plagiarism. If copying is detected, all those involved will lose credit, irrespective of who copied from whom.

Question 1. Show what this program does, step by step, as in Section 3.2 of the web notes.

```
#include <stdio.h>
int main()
{ int i; double x;
    x = 1;
    for (i=0; i<3; ++i)
    { x = x+x;
    }
    printf("Now x is %f\n", x);
}</pre>
```

Answer.

```
x = 1
Preparation i = 0
condition: i<3? yes
x=1+1=2
next i = 1
condition i<3? yes
x = 2+2 = 4
next i=2
condition i<3? yes
x = 4+4 = 8
next i = 3
condition i<3? no
prints Now x is 8.000000
```

Question 2. If you can, compile and run this program, show the output, and say which (k, x) is correct.

If you do not yet have ready access to a C compiler, say what the final values of k and x ought to be.

#include <stdio.h>
int main()
{ int k; double x;

```
k = 1; x = 1;
while ( k > 0 )
{ k = k+k;
    x = x+x;
}
printf("k %d x %f\n", k, x);
}
```

Answer.

```
k -2147483648 x 2147483648.000000
```

x is correct. One expects the answer 2^{31} .

Question 3. In the following code, when n is 15, the final value of ell is 3. (i) What gets printed if you change n from 15 to 100? (ii) For general $n \ge 1$, what gets printed as ell? That is, the final value of ell is a certain function of n: what is that function?

```
#include <stdio.h>
int main()
{
  int n, nn, ell;
  n = 15; nn = n; ell = 0;
  while (nn > 1)
  \{ ++ \text{ ell}; \text{ nn} = \text{nn}/2; \}
  }
  printf("n is %d, ell is %d\n", n, ell );
}
   Answer.
(i) n is 100, ell is 6
   (ii) In general, ell is, finally, |\log_2(n)|.
   Question 4. Show the output to the code below, and explain it.
#include <stdio.h>
int main()
{
  int m; double x;
  m = 17; x = 17;
  printf("m is %d, m - 3 * (m/3) is %d\n", m, m - 3 * (m/3) );
  printf("x is %f, x - 3 * (x/3) is %f\n", x, x - 3 * (x/3) );
  m = -17; x = -17;
  printf("m is %d, m - 3 * (m/3) is %d\n", m, m - 3 * (m/3) );
  printf("x is %f, x - 3 * (x/3) is %f\n", x, x - 3 * (x/3) );
}
```

Answer.

```
m is 17, m - 3 * (m/3) is 2
x is 17.000000, x - 3 * (x/3) is 0.000000
m is -17, m - 3 * (m/3) is -2
x is -17.000000, x - 3 * (x/3) is 0.000000
```

(i) $m - 3(m/3) = m \mod 3 = 2$. (ii) x - 3(x/3) = 0. (iii) m/3 = -5 (rounded up) and m - 3(m/3) = -17 + 15 = -2. (iv) x - 3(x/3) = 0.

Question 5. (i) Show what this program does, step by step, as in Section 3.2 of the web notes. (ii) For general n, what do the answers signify? Details are not required.

#include <stdio.h>

```
int main()
{ int x, i, n;
  x = 0; i=0; n = 3;
  while (x < n)
  \{ x = x + 2*i + 1; \}
    i = i+1;
  }
  if (x == n)
    printf("Yes\n");
  else
    printf("No\n");
}
  Answer.
(i)
i, x = 0; n = 3;
x<n? yes
x = x + 2*i + 1 = 1
i = i+1 = 2
```

```
i = i+1 = 2
x<n? yes
x = x + 2*i + 1 = 1 + 2 + 1 = 4
x<n ? no
loop ends
x==n? 4 == 3? no
prints No</pre>
```

(ii) Yes if n is a perfect square, No otherwise.