

QUESTIONS ON LECTURES 11-13

- a) Associate **semantic rules**, using only **synthesised attributes**, to the rules of the following grammar to **calculate the value of the entire expression**. The terminal symbol digit may have any value from 0 to 9.

$G \rightarrow E$
 $E \rightarrow E + T$
 $E \rightarrow T$
 $T \rightarrow T * F$
 $T \rightarrow F$
 $F \rightarrow (E)$
 $F \rightarrow \text{digit}$

State any assumptions you make.

(5 marks)

- b) Draw the **Control Flow Graph** for the following code fragment:

```
m=0; n=0
L1: if (m>n) goto L2
s=A[1]
do {
  n++
  if (n>A[n])
    break // exits do ... while loop
  n=n+3
}
while (s>n)
m=m+A[m]
goto L1
L2: return
```

(5 marks)

- c) How many **activation records** exist in the stack when the execution of the code reaches the printf statement for the first time?

```
void A(int m) { if(m>0) B(m); else C(-m); }
void B(int n) { if(n>1) B(n-1); else D(1); }
void C(int x) { if(x%2!=0) A(x-1); else D(2); }
void D(int y) { if(y<3) printf("%d\n",y); else printf("h\n"); }
main() { A(-3); }
```

Draw the call graph.

(5 marks)

- d) A certain compiler for a simple language implements a **2048-entry long symbol table** in order to store information about program names. The **hash function**, which is used to map names onto the symbol table, is based on the following algorithm.

Every name is **divided into four-character chunks** (starting from the left). Each chunk is then converted to a **32-bit long integer** by concatenating the ASCII value of **each character (8 bits)**, starting from the left (this is equivalent to the value $256^3a_1 + 256^2a_2 + 256a_3 + a_4$, where a_1, a_2, a_3, a_4 , are the ASCII value of each character in the chunk, with a_1 being the leftmost character). If a particular chunk contains **fewer than 4 characters** it is assumed that the characters **missing** (to the right) have an ASCII value of **zero**. A **bitwise exclusive-or** operation is then performed between all the chunks; this starts with the 1st and 2nd chunk (counting from the left); the result is then XOR-ed with the 3rd chunk, and so on. The resulting integer after the final XOR operation is **divided by 2048**; the **remainder** of the division is then used to map the **name** onto the symbol table.

Would this hash function be a good choice? Justify your answer.

(5 marks)