SSN COLLEGE OF ENGINEERING, KALAVAKKAM

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Compiler Design Lab – CS6612

PROGRAMMING ASSIGNMENT 9 – Simulation of storage allocation strategy using stack

---

Date of Assignment: 06.03.2017 **Due Date: 20.03.17 & 22.03.17**

Consider the following source code snippet to print the square root of a number involving the following function calls.

```
Int main()
{
  int temp;
  temp=add(a,b);

}

add(x,y)
{
 return sub(x,y);
}

sub(x,y)
{
 return mul(x,y);
}

mul(x,y)
{
 return div(x,y);
}

div(x,y)
{
 return sqrt(x);
}
```

Associate addresses for every statement in the program. Assume a starting address of the program. Following the start statement, every statement must have address in the increments of 20.

Scan this source code for the function declarations to record the number and type of the parameters. Write a program to simulate the storage allocation strategy by implementing stack

to facilitate pushing and popping of activation records during function calls. Generate the random starting memory address and use this address to initialize the stack pointer. Activation record should be the member of the stack. The stack pointer needs to be updated based on the sizeof the activation record. The activation record is a structure that contains the return address, number of parameters and types of parameters. The example of activation record is shown below.

Return address – 140
2
int, int

Print the flow of pushing and popping of activation records and their contents during function call and exit respectively. The sample source program and output are as follows:

Sample Source Program:

```
100     int add(int,int);
120     int sub(float,float);
140     int main()
        {
160       int a=10,b=45, sum;
180       float x=10.9,y=45.2, difference;
200       sum=add(a,b);
220       difference=sub(x,y);
240       return 0;
        }
```

Sample Output:

Initial Stack pointer address → 2000

Call – Add function – Pushed Activation Record

Return address – 220
No. of parameters – 2
Types of parameters – int, int

Stack pointer address → 2018

Exit – Add function - Popped Activation Record

Return address – 220
No. of parameters – 2
Types of parameters – int, int

Stack pointer address → 2000

Call – Sub function - Pushed Activation Record

Return address – 240
No. of parameters – 2
Types of parameters – float, float

Stack pointer address → 2018

Exit – Sub function - Popped Activation Record

Return address – 240
No. of parameters – 2
Types of parameters – float, float

Stack pointer address → 2000