

Sessional Assignment
Software Project Management

Marks: 20

NAME: SHEHARYAR TAHIR
ID# 13484

1. Implement cocomo II model in any Language (C++, c#, Java, Any other)

ANSWER:

CODE IN C++:

```
#include<cstdlib>
#include<iostream>
#include<fstream>
#include<string.h>
#include<cmath>
using namespace std;
int matsin(char *str){
    int len=strspn(str,"//");
    return len;
}
int matmul(char *str){
    int len=strspn(str,"/*");
    return len;
}

int countExpComm(char *filename){
```

```
int SIZE=100;
char str[SIZE];
int ln=0;
int lbegcomm=0;
int lendcomm=0;
bool firstmat=false;
bool secondmat=false;
ifstream in(filename,ios::in);
    if(!in){
        cout<<"Could not open the file!";
        return(100);
    }
while(in.getline(str,SIZE)){
    ln++;
    //excluding single and multiple lines
    if (matsin(str)==2)
        ln--;
    else if(matmul(str)==2){
        if(firstmat==false){
            firstmat=true;
            lbegcomm=ln;
        }
        else{
            secondmat=true;
            lendcomm=ln;
        }
    }
}
```

```

    }
}
if(firstmat==true && secondmat==true)
{

    ln=ln-(lendcomm-lbegcomm+1);
    firstmat=false;
    secondmat=false;
}

}

in.close();
return ln;
}

//Count blank lines
int countBlankL(char *filename){
    char ch;
    char prech='\0';
    int lblank=0;
    ifstream in(filename,ios::in);
    if(!in){
        cout<<"Could not open the file!";
        return(100);
    }
}

```

```

while(in.get(ch)){
    if(ch=='\n' && prech=='\n')
        lblank++;
    prech=ch;
}

```

```

in.close();
return lblank;
}

```

//calculate lloc

```

int lloc(char *filename){
return(countExpComm(filename)-countBlankL(filename));
}

```

```

void cocomo(int loc){

```

```

    float eaf, e, effort, duration, se, person, ksloc;

```

```

    float cd1, cd2, cd3, cd4, cd5, cd6, cd7, cd8, cd9, cd10, cd11, cd12, cd13, cd14,
    cd15, cd16, cd17;

```

```

    float sd1, sd2, sd3, sd4, sd5;

```

```

    cout<<"CALCULATE EFFORT, DURATION, AND PEOPLE REQUIRED TO
COMPLETE A SOFTWARE PROJECT"<<endl;

```

```

    cout<<"=====COCOMO
II===== "<<endl;

```

```

    cout<<"=====Number of Lines Of Code
(LOC)===== "<<endl;

```

```

    cout<<"LOC = "<<loc<<endl;

```

```

cout<<"=====CULCULATE
EAF===== "<< endl;

cout<<"EAF = product (All 17 Cost Drivers)"<<endl;

cout<<"=====Please input all 17 cost driver ===== "<<endl;

cout<<"1. Programmer capability: ";cin>>cd1;
cout<<"2. Required system Reliability: ";cin>>cd2;
cout<<"3. Complexity of system modules: ";cin>>cd3;
cout<<"4. Extent of documentation required: ";cin>>cd4;
cout<<"5. Size of database used: ";cin>>cd5;
cout<<"6. Required percentage of reusable components: ";cin>>cd6;
cout<<"7. Execution time constraint: ";cin>>cd7;
cout<<"8. volatility of development platform: ";cin>>cd8;
cout<<"9. Memory constraints: ";cin>>cd9;
cout<<"10. Capability of project analysts: ";cin>>cd10;
cout<<"11. Personnel continuity: ";cin>>cd11;
cout<<"12. Programmer experience in project domain: ";cin>>cd12;
cout<<"13. Analyst experience in project domain: ";cin>>cd13;
cout<<"14. Language and tool experience: ";cin>>cd14;
cout<<"15. Use of software tools: ";cin>>cd15;
cout<<"16. Development schedule compression: ";cin>>cd16;
cout<<"17. Extent of multisite working and quality of inter-site communications:
";cin>>cd17;

eaf=(cd1*cd2*cd3*cd4*cd5*cd6*cd7*cd8*cd9*cd10*cd11*cd12*cd13*cd14*cd1
5*cd16*cd17);

cout<<"==> EAF = "<<eaf<<endl;

```

```

cout<<"=====CULCULATE
E===== "<< endl;

cout<<"E = 1.01 + sum(All 5 Scale Drivers)"<<endl;

cout<<"=====Please input all 5 Scale Driver ===== "<<endl;

cout<<"1. Precedentedness: ";cin>>sd1;

cout<<"2. Development Flexibility: ";cin>>sd2;

cout<<"3. Architeture/Rise Resolution: ";cin>>sd3;

cout<<"4. Team Cohesion: ";cin>>sd4;

cout<<"5. Process Maturity: ";cin>>sd5;

e= 1.01 + (sd1+sd2+sd3+sd4+sd5);

cout<<"=> E = "<<e<<endl;

cout<<"=====CULCULATE
EFFFORT===== "<< endl;

cout<<"Effort = 2.94 * EAF * (KSLOC)^E"<<endl;

cout<<"Where KSLOC = LOC/1000"<<endl;

ksloc=loc/1000;

cout<<"=>KSLOC = "<<ksloc<<endl;

effort=2.94*eaf*pow(ksloc,e);

cout<<"Effort = "<<effort<<" person-months"<<endl;

cout<<"=====CULCULATE
DURATION===== "<< endl;

cout<<"Duration = 3.67 * (Effort)^SE"<<endl;

cout<<"Where SE = 0.28 + 0.2 * (E - 1.01)"<<endl;

se= 0.28 + (0.2 * (e - 1.01));

cout<<"=> SE = "<<se<<endl;

duration = 3.67 * pow(effort,se);

```

```

cout<<"Duration = "<<duration<<" months"<<endl;
cout<<"=====CULCULATE NUMBER OF
PEOPLE===== "<< endl;
cout<<"Person = Effort/Duration"<<endl;
person=effort/duration;
cout<<"Person = "<<person<<" people"<<endl;

cout<<"=====END=====
====="<< endl;
}

int main(){
    char filename[100];
    cout<<"Enter file name:";
    gets(filename);
    cocomo(lloc(filename));
    system("PAUSE");
    return 0;
}

```

OUTPUT:

```
C:\Users\User\Desktop\cocomo 2 13484 sheryar.exe
Enter file name:cocomo 2 shery
Could not open the file!Could not open the file!CALCULATE EFFORT, DURATION, AND PEOPLE REQUIRED TO COMPLETE A SOFTWARE PROJECT
=====COCOMO II=====
=====Number of Lines Of Code (LOC)=====
LOC = 0
=====CULCULATE EAF=====
EAF = product (All 17 Cost Drivers)
=====Please input all 17 cost driver =====
1. Programmer capability: 11
2. Required system Reliability: 12
3. Complexity of system modules: 13
4. Extent of documentation required: 14
5. Size of database used: 15
6. Required percentage of reusable components: 17
7. Execution time constraint: 19
8. volatility of development platform: 21
9. Memory constraints: 23
10. Capability of project analysts: 25
11. Personnel continuity: 27
12. Programmer experience in project domain: 29
13. Analyst experience in project domain: 31
14. Language and tool experience: 33
15. Use of software tools: 35
16. Development schedule compression: 37
17. Extent of multisite working and quality of inter-site communications: 39
==> EAF = 5.68588e+022
=====CULCULATE E=====
E = 1.01 + sum(All 5 Scale Drivers)
=====Please input all 5 Scale Driver =====
1. Precedentedness: 1
2. Development Flexibility: 4
3. Architecture/Rise Resolution: 7
4. Team Cohesion: 9
5. Process Maturity: 11
==> E = 33.01
=====CULCULATE EFFORT=====
Effort = 2.94 * EAF * (KSLOC)^E
Where KSLOC = LOC/1000
==>KSLOC = 0
Effort = 0 person-months
=====CULCULATE DURATION=====
Duration = 3.67 * (Effort)^SE
Where SE = 0.28 + 0.2 * (E - 1.01)
==> SE = 6.68
Duration = 0 months
=====CULCULATE NUMBER OF PEOPLE=====
Person = Effort/Duration
Person = nan people
=====END=====
Press any key to continue . . .
```