Chittagong University of Engineering & Technology

Department of **Computer science & Engineering**

A PROJECT OF MAKING A CALCULATOR USING C++(OBJECT ORIENTED PROGRAMMING)

Course number: CSE- 144

Course title: Object Oriented Programming (Sessional)

Experiment number: 09

Date of submission: January 17,2012

Remarks

Name of the student: Emtiaj Hasan

Student's ID number: 1004050

Level: I

Term: II

Section: A

Group: A-2

OBJECTIVE

The term project is to make a calculator by using basic feature of Object Oriented Programming .

```cpp
// EMTIAJ HASAN 1004050
#include<iostream>
#include<cstring>
using namespace std;

const double   PI=3.14159265358979323846264338327950288419716939937510;
const int  SIZE=120;
char s1[SIZE],s2[SIZE],tmp[SIZE],res[SIZE],final[SIZE],summa[SIZE];
char str[50][SIZE];

class base
{
      public:
      base();
      void  addition( void  );
      void  subtraction( void  );
      void  multiplication( void  );
      void  division( void  );
      double   abs( double   );
      void  sqrt  ( void  );
      double   sqrt  ( double   );
      void  pow ( void  );
      int  square  ( int );
      int  pow ( int , int );
      void  inverse  ( void  );
      void  quad_eqn( void  );
      void  l_eqn_2( void  );
      void  l_eqn_3( void  );
};
base::base()
{
      cout<<"\n    HELLO, THIS IS EMTIAJ HASAN. I HAVE MADE A SIMPLE CALCULATOR USING C++.\n\n";
      cout<<"\t1. ADDITION";
      cout<<"\t2. SUBTRACTION";
      cout<<"\t3. MULTIPLICATION\n\n";
      cout<<"\t4. DIVISION";
      cout<<"\t5. FACTORIAL";
      cout<<"\t6. COMBINATION\n\n";
      cout<<"\t7. PERMUTATION";
      cout<<"\t8. GCD LCM";
      cout<<"\t9. X^n\n\n";
      cout<<"\t10.INVERSE";
      cout<<"\t11.SQUARE ROOT\n\n";
      cout<<"\t12.SOLVING QUADRATIC EQUATION\n\n";
      cout<<"\t13.SOLVING A SET OF TWO LINEAR EQUATION\n\n";
      cout<<"\t14.SOLVING A SET OF THREE LINEAR EQUATION\n\n";
      cout<<"\t15.NATURAL LOGARITHM";
      cout<<"\t16.10 BASE LOGARITHM";
      cout<<"\t17.EXPONENTIAL\n\n";
      cout<<"\t18.sin(x)";
      cout<<"\t19.cos(x)";
      cout<<"\t20.tan(x)";
      cout<<"\t21.cot(x)\n\n";
      cout<<"\t0.EXIT\n\n";
      cout<<"\n\tENTER CHOICE ";
}
//addition
void   base:: addition( void  )
{
      int  l1, l2, i, j, temp, m, n, sum, hold;
      int  k, len;

      cout<<"HOW MANY NUMBERS YOU WOULD LIKE TO INPUT? ";
      cin>>len;
      cout<<"INPUT NUMBERS (PRESS ENTER AFTER EACH NUMBER)\n";
      for( k=0;k<len;k++)
         cin>>str[k];

      strcpy( summa, "0");
      for( k=0;k<len;k++){
         strcpy( s1, str[k]);
```

```
            strcpy(s2,summa);
            l1=strlen(s1);
            l2=strlen(s2);
            if(l2>l1){
                temp=l1;
                l1=l2;
                l2=temp;
                strcpy(tmp,s1);
                strcpy(s1,s2);
                strcpy(s2,tmp);
            }
            m=l1-1;
            n=l2-1;
            for(i=0,hold=0;i<l1;i++,m--,n--){
                if(n>=0)sum=(s1[m]-'0')+(s2[n]-'0')+hold;
                else sum=(s1[m]-'0')+hold;
                res[i]=sum%10+'0';
                hold=sum/10;
            }
            if(hold) res[i++]=hold+'0';
            for(j=i-1,i=0;j>=0;j--,i++)
                final[i]=res[j];

            final[i]='\0';
            strcpy(summa,final);
        }
        cout<<"THE ADDITION IS "<<final<<endl;
    }
    //subtraction
    void base:: subtraction(void)
    {
        int i,j,l1,l2,m,n,temp,hold,diff;
        char sign='0';
        cout<<"ENTER TWO NUMBER FOR DOING SUBTRACTION\n";
        cout<<"PRESS ENTER AFTER EACH NUMBER\n";
        cin>>s1>>s2;
        l1=strlen(s1);
        l2=strlen(s2);
        if(l1<l2){
            temp=l1;
            l1=l2;
            l2=temp;
            strcpy(tmp,s1);
            strcpy(s1,s2);
            strcpy(s2,tmp);
            sign='1';
        }
        if(l1==l2){
            if(strcmp(s1,s2)<0){
                strcpy(tmp,s1);
                strcpy(s1,s2);
                strcpy(s2,tmp);
                sign='1';
            }
        }
        m=l1-1;
        n=l2-1;
        for(i=0,hold=0;i<l1;i++,m--,n--){
            if(n>=0)diff=(s1[m]-'0')-(s2[n]-'0')-hold;
            else diff=(s1[m]-'0')-hold;
            if(diff<0){
                hold=1;
                res[i]=diff+'0'+10;
            }
            else{
                res[i]=diff+'0';
                hold=0;
            }
        }
        for(i=l1-1;i>0;i--){
            if(res[i]!='0') break;
        }
```

```
144         res[i+1]='\0';
145         if(sign=='1') cout<<"THE SUBTRACTION IS -";
146         else if(sign=='0')cout<<"THE SUBTRACTION IS ";
147         for(j=i,i=0;j>=0;j--,i++)
148             cout<<res[j];
149 
150         cout<<endl;
151     }
152     //multiplication
153     void base:: multiplication(void)
154     {
155         int a,b,sum;
156         cout<<"ENTER TWO NUMBERS FOR DOING MULTIPLICATION\n";
157         cin>>a>>b;
158         if(a!=0&&b!=0){
159             if(a%2==0)sum=0;
160             else sum=b;
161             while(a!=1){
162                 a/=2;
163                 b=b*2;
164                 if(a%2!=0) sum+=b;
165             }
166             cout<<"THE PRODUCT IS "<<sum<<endl;
167         }
168         else cout<<"THE PRODUCT IS 0\n";
169     }
170     //division
171     void base:: division(void)
172     {
173         int divisor,mod,i,j,k,q,emti;
174         cout<<"ENTER TWO NUMBERS FOR DOING DIVISION\n";
175         cout<<"PRESS ENTER AFTER EACH NUMBER\n";
176         cin>>s1>>divisor;
177         if((s1[0]=='0'&&s1[1]=='\0')&&divisor==0) cout<<"INDETERMINATE\n";
178         else if(divisor==0) cout<<"UNDEFINED\n";
179         else{
180             for(i=0,j=0,k=0;s1[i];i++){
181                 emti=(s1[i]-'0')+k;
182                 q=emti/divisor;
183                 mod=emti%divisor;
184                 k=mod*10;
185                 if(q) final[j++]=q+'0';
186             }
187             final[j]='\0';
188             if(final[0]=='\0') cout<<"THE QUOTIENT IS 0 MOD IS "<<mod<<endl;
189             else cout<<"THE QUOTIENT IS "<<final<<", MOD IS "<<mod<<endl;
190         }
191     }
192     //abs()
193     double base::abs(double x)
194     {
195         if(x<0) x=-x;
196         return x;
197     }
198     void base::sqrt(void)
199     {
200         double num;
201         cout<<"ENTER A NUMBER(REMEMBER SQUAREROOT OF A NEGETIVE NUMBER IS
    IMAGINARY)\n";
202         cin>>num;
203         cout<<"SQRT("<<num<<") = "<<sqrt(num)<<endl;
204     }
205     //sqrt
206     double base::sqrt(double num)
207     {
208         double guess=1.0;
209         const double  acc = 0.00001;
210         while(abs(guess*guess-num)>=acc)
211             guess=(num/guess+guess)/2.0;
212 
213         return guess;
214     }
```

```
215  void base::pow(void)
216  {
217      int bas,power;
218      cout<<"ENTER BASE & POWER RESPECTIVELY ";
219      cin>>bas>>power;
220      cout<<bas<<"^"<<power<<" = "<<pow(bas,power)<<endl;
221  }
222  int base::square(int n)
223  {
224      return n*n;
225  }
226  //x^n
227  int base::pow(int base,int power)
228  {
229      if(power==0) return 1;
230      else if(power%2==0) return square(pow(base,power/2));
231      else return base*(pow(base,power-1));
232  }
233  //a^-1
234  void base::inverse(void)
235  {
236      double inv;
237      cout<<"ENTER A NUMBER TO INVERSE IT ";
238      cin>>inv;
239      cout<<"1/"<<inv<<" = "<<(1/inv)<<endl;;
240  }
241  //quad
242  void base::quad_eqn(void)
243  {
244      int a,b,c;
245      double dis,r,x1,x2;
246      cout<<"QUADRATIC EQUATION IS ax^2+bx+c=0\n";
247      cout<<"ENTER CO-EFFICIENT OF of x^2, x & CONSTANT.\n";
248      cin>>a>>b>>c;
249      dis=b*b-4*a*c;
250      if(dis<0) cout<<"SINCE DISCRIMINANT OF THIS EQUATION IS NEGETIVE, THE ROOTS
     ARE IMAGINARY.\n";
251      else
252      {
253          r=sqrt(dis);
254          x1=(-b+r)/(2*a);
255          x2=(-b-r)/(2*a);
256          cout<<"X1 = "<<x1<<endl<<"X2 = "<<x2;
257      }
258  }
259  // solve x,y
260  void base::l_eqn_2(void)
261  {
262      int a1,a2,b1,b2,c1,c2,det,dx,dy;
263      double x,y;
264      cout<<"A SET OF TWO LINEAR EQUATION IS "<<endl;
265      cout<<"\ta1x + b1y = c1\n\ta2x + b2y = c2\n";
266      cout<<"ENTER CO-EFFICIENT OF x, y & CONSTANT.\n";
267      cin>>a1>>b1>>c1>>a2>>b2>>c2;
268      det=a1*b2-a2*b1;
269      if(det==0) cout<<"ERROR\n";
270      else{
271          dx=(c1*b2-b1*c2);
272          dy=(c2*a1-a2*c1);
273          x=(double)dx/(double)det;
274          y=(double)dy/(double)det;
275          cout<<"\nX= "<<dx<<"/"<<det<<" = "<<x<<endl;
276          cout<<"\nY= "<<dy<<"/"<<det<<" = "<<y<<endl;
277      }
278  }
279  //solve x,y,z
280  void base::l_eqn_3(void)
281  {
282      int a1,a2,a3,b1,b2,b3,c1,c2,c3,d1,d2,d3,det,dx,dy,dz;
283      double x,y,z;
284      cout<<"A SET OF THREE LINEAR EQUATION IS "<<endl;
285      cout<<"\ta1x + b1y + c1z = d1\n\ta2x + b2y + c2z = d2\n\ta3x + b3y + c3z =
```

```
    d3\n";
286         cout<<"ENTER CO-EFFICIENT OF x, y, z & CONSTANT.\n";
287         cin>>a1>>b1>>c1>>d1>>a2>>b2>>c2>>d2>>a3>>b3>>c3>>d3;
288         det= a1*(b2*c3-c2*b3)-b1*(a2*c3-c2*a3)+c1*(a2*b3-a3*b2);
289         if(det==0) cout<<"ERROR\n";
290         else{
291             dx=(d1*(b2*c3-c2*b3)-b1*(d2*c3-c2*d3)+c1*(d2*b3-d3*b2));
292             dy=(a1*(d2*c3-c2*d3)-d1*(a2*c3-c2*a3)+c1*(a2*d3-a3*d2));
293             dz=(a1*(b2*d3-d2*b3)-b1*(a2*d3-d2*a3)+d1*(a2*b3-a3*b2));
294             x=(double)dx/(double)det;
295             y=(double)dy/(double)det;
296             z=(double)dz/(double)det;
297             cout<<"\nX= "<<dx<<"/"<<det<<" = "<<x<<endl;
298             cout<<"\nX= "<<dx<<"/"<<det<<" = "<<y<<endl;
299             cout<<"\nX= "<<dx<<"/"<<det<<" = "<<z<<endl;
300         }
301     }
302
303     //derived 1
304     class derived1 : public base
305     {
306         public:
307         void ln(void);
308         double ln(double);
309         void log10(void);
310         void exponential(void);
311         double factorial(int);
312         void sine(void);
313         void cosine(void);
314         void tangent(void);
315         void cotangent(void);
316         double sine(double);
317         double cosine(double);
318     };
319     //e^x
320     void derived1::exponential(void)
321     {
322         double i,j,e,sum,g,t_sum,t_e;
323         cout<<"ENTER A NUMBER ";
324         cin>>e;
325         t_e=e;
326         for(i=1,j=1,sum=0;;i++,j=t_e){
327             e=e*j;
328             g=factorial(i);
329             t_sum=e/g;
330             if(t_sum<=0.001){
331                 sum=t_sum+sum;
332                 cout<<"e^"<<t_e<<" = "<<sum+1<<endl;;
333                 break;
334             }
335             else sum=t_sum+sum;
336         }
337     }
338     void derived1::ln(void)
339     {
340         double x;
341         cout<<"ENTER A NUMBER ";
342         cin>>x;
343         if (x<=0){
344             if(x==-1) cout<<"ln(-1) = i*PI\n";
345             else cout<<"MATH ERROR.\n";
346         }
347         else cout<<"ln("<<x<<") = "<<ln(x)<<endl;
348     }
349     //ln(x)
350     double derived1::ln(double x)
351     {
352         int i,j;
353         double e,r,sum,deno,neu,f_sum;
354         sum=0.0;
355         neu=x-1;
356         deno=x+1;
```

```
357         e=neu/deno;
358         for (i=1;i<=25;i=i+2){
359             r=1;
360             f_sum=1/i;
361             for(j=1;j<=i;j++)
362                 r*=e;
363 
364             sum+=r/i;
365         }
366         f_sum=sum*2;
367         return f_sum;
368     }
369     //log10(x)
370     void derived1::log10(void)
371     {
372         double n;
373         cout<<"ENTER NUMBER ";
374         cin>>n;
375         if(n<0) cout<<"MATH ERROR\n";
376         else cout<<"log10("<<n<<") = "<<ln(n)/2.3025850929940<<endl;
377     }
378     //factorial
379     double derived1::factorial(int n)
380     {
381         double fact=1;
382         for(int i=1;i<=n;i++)
383             fact*=(double)i;
384 
385         return fact;
386     }
387     void derived1::cosine(void)
388     {
389         double theta,t_theta,res;
390         cout<<"ENTER A NUMBER ";
391         cin>>theta;
392         t_theta=(int)theta;
393         if(t_theta<0) theta=(-1)*theta;
394         res=cosine(theta);
395         if(t_theta<0)cout<<"\ncos(-"<<theta<<") = "<<res<<endl;
396         else cout<<"\ncos("<<theta<<") = "<<res<<endl;
397     }
398     void derived1::sine(void)
399     {
400         double deg,t_d,res;
401         cout<<"ENTER A NUMBER ";
402         cin>>deg;
403         t_d=(int)deg;
404         if(t_d<0) deg=(-1)*deg;
405         res=sine(deg);
406         if(t_d<0)cout<<"sin(-"<<deg<<") = -"<<res<<endl;
407         else cout<<"sin("<<deg<<") = "<<res<<endl;}
408     void derived1::tangent(void)
409     {
410         double k,l,m;
411         cout<<"ENTER A NUMBER\n";
412         cin>>k;
413         if((int)k==90) cout<<"UNDEFINED\n";
414         else{
415             l=sine(k);
416             m=cosine(k);
417             cout<<"tan("<<k<<") = "<<l/m<<endl;
418         }
419     }
420     void derived1::cotangent(void)
421     {
422         double k,l,m;
423         cout<<"ENTER A NUMBER\n";
424         cin>>k;
425         if((int)k==0) cout<<"UNDEFINED\n";
426         else{
427         l=sine(k);
428         m=cosine(k);
```

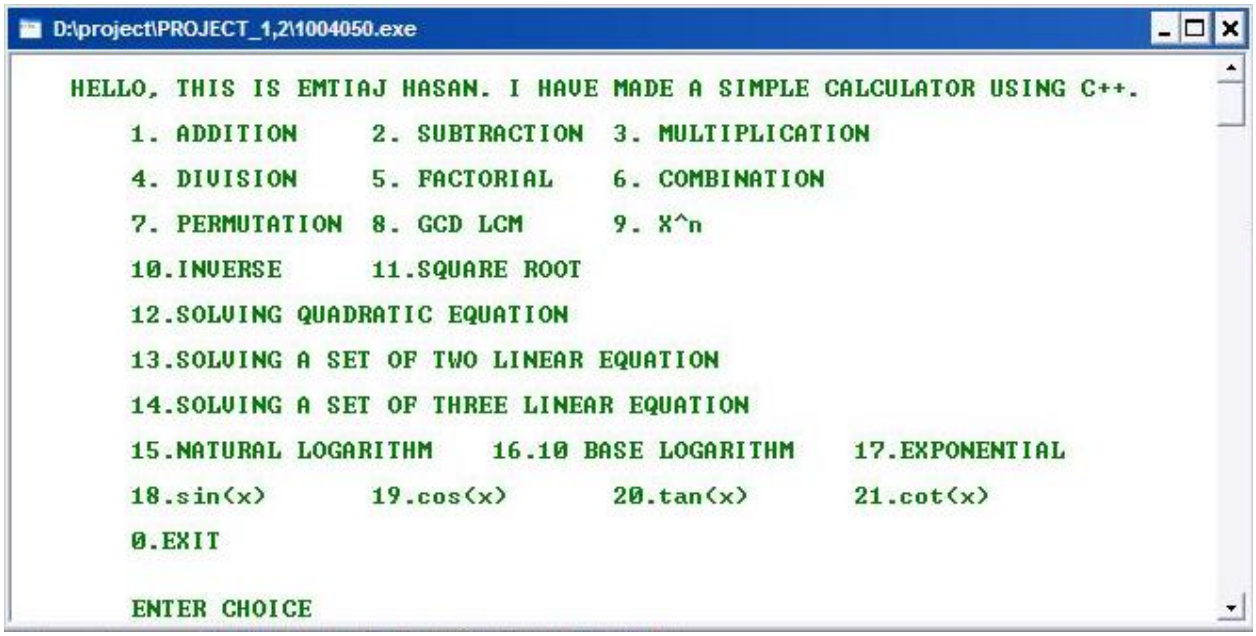
```
429         cout<<"cot("<<k<<") = "<<m/l<<endl;
430         }
431     }
432     double derived1::sine(double deg)
433     {
434         const double ACC=0.001;
435         double res,rad,term;
436         int i,j,k;
437         rad=(PI*deg)/180;
438         res=rad;
439         term=rad;
440         i=2;
441         while(1){
442             if(term<0){
443                 if(-term<ACC)break;
444             }
445             else{
446                 if(term<ACC)break;
447             }
448             term*=-((rad/i)*(rad/(i+1)));
449             res+=term;
450             i+=2;
451         }
452         return res;
453     }
454     double derived1::cosine(double deg)
455     {
456         double sum=0,res=1,rad;
457         int j=0;
458         rad=deg*(PI/180);
459         for(int i=1;i<=20;i++){
460             if(i==1)sum=sum+1;
461             else{
462                 if(i%2==0){
463                     for(int k=1;k<=j;k++)
464                         res=res*k;
465 
466                     sum=sum-rad*rad/res;
467                     res=1;
468                 }
469                 else{
470                     for(int k=1;k<=j;k++)
471                         res=res*k;
472 
473                     sum=sum+rad*rad/res;
474                     res=1;
475                 }
476             }
477             j=j+2;
478         }
479         return sum;
480     }
481     //derived2
482     class derived2 : public derived1
483     {
484         public:
485         void facto(void);
486         void combination(void);
487         void permutation(void);
488         int combination(int,int);
489         int permutation(int,int);
490         void g_l(void);
491         int GCD(int,int);
492         int LCM(int,int);
493     };
494     void derived2::combination(void)
495     {
496         int n,r;
497         cout<<"ENTER TWO NUMBER FOR DOING COMBINATION"<<endl;
498         cin>>n>>r;
499         if(r>n) cout<<"MATH ERROR\n";
500         else cout<<n<<"C"<<r<<" = "<<combination(n,r)<<endl;
```

```
501   }
502
503   void derived2::permutation(void)
504   {
505       int n,r;
506       cout<<"ENTER TWO NUMBER FOR DOING PERMUTATION"<<endl;
507       cin>>n>>r;
508       if(r>n) cout<<"MATH ERROR\n";
509       else cout<<n<<"P"<<r<<" = "<<permutation(n,r)<<endl;
510   }
511   //combination
512   int derived2::combination(int n,int r)
513   {
514       int j;
515       int c;
516       if(r>n/2) r=n-r;
517       for(j=0,c=1;j<r;j++){
518           c=c*(n-j);
519           c=c/(1+j);
520       }
521       return c;
522   }
523   //permutation
524   int derived2::permutation(int n,int r)
525   {
526       int p;
527       p=combination(n,r)*(int)factorial(r);
528       return p;
529   }
530   void derived2::facto(void)
531   {
532       int n;
533       cout<<"ENTER A NUMBER FOR DOING FACTORIAL ";
534       cin>>n;
535       if(n<0&&n%2==0) cout<<"FACTORIAL OF NEGETIVE EVEN NUMBER IS -INFINITY"<<endl;
536       else if(n<0&&n%2!=0) cout<<"FACTORIAL OF NEGETIVE ODD NUMBER IS +INFINITY"<<
      endl;
537       else cout<<n<<"! = "<<factorial(n)<<endl;
538   }
539   void derived2::g_l(void)
540   {
541       int i,len,gcd,lcm,data[100];
542       cout<<"HOW MANY NUMBERS ";
543       cin>>len;
544       for(i=0;i<len;i++)
545           cin>>data[i];
546
547       gcd = GCD( data[0], data[1] ) ;
548       lcm=(data[0]*data[1])/gcd;
549       for( i = 2 ; i < len ; i++ ){
550           gcd = GCD( gcd, data[i] ) ;
551           lcm = LCM( lcm, data[i]);
552       }
553       cout<<"GCD OF THOSE INPUT IS "<<gcd<<endl;
554       cout<<"LCM OF THOSE INPUT IS "<<lcm<<endl;
555
556   }
557   //gcd
558   int derived2::GCD(int first,int second)
559   {
560       int mod ;
561       while( second != 0 ){
562           mod = first % second ;
563           first=second;
564           second=mod;
565       }
566       return first;
567   }
568   //lcm
569   int derived2::LCM( int i, int j )
570   {
571       return i*j/GCD(i, j) ;
```

```
}
//main()
int main()
{
    derived2 A;
    int choice;
    char ch;
    while(cin>>choice&&choice){
        switch(choice){
            case 1:
                A.addition();
                break;
            case 2:
                A.subtraction();
                break;
            case 3:
                A.multiplication();
                break;
            case 4:
                A.division();
                break;
            case 5:
                A.facto();
                break;
            case 6:
                A.combination();
                break;
            case 7:
                A.permutation();
                break;
            case 8:
                A.g_l();
                break;
            case 9:
                A.pow();
                break;
            case 10:
                A.inverse();
                break;
            case 11:
                A.sqrt();
                break;
            case 12:
                A.quad_eqn();
                break;
            case 13:
                A.l_eqn_2();
                break;
            case 14:
                A.l_eqn_3();
                break;
            case 15:
                A.ln();
                break;
            case 16:
                A.log10();
                break;
            case 17:
                A.exponential();
                break;
            case 18:
                A.sine();
                break;
            case 19:
                A.cosine();
                break;
            case 20:
                A.tangent();
                break;
            case 21:
                A.cotangent();
                break;
```

```
644             }
645             cout<<"\nPRESS y IF YOU WOULD LIKE TO CONTINUE, ELSE n\n";
646             cin>>ch;
647             if(ch!='n') derived2 A;
648             else break;
649         }
650         cout<<"\n\n\t\tTHANK YOU FOR USING MY CALCULATOR\n\n";
651         return 0;
652     }
653
```

OUTPUT



DISCUSSION

In this term, we were taught Object Oriented Programming through C++. As this is the eleventh hour of this term, we were given a project. In this task I use some features of C++ (OOP) such as constructor,inheritence, polymorphism (function overloading),etc. I divide this code into small part through function. Some function is set in base Class and some in derived Class. Derived class is inherited Publicly. So I create an object of derived class and through this object everything has been done. Though it is a simple calculator, I hope I could make a better one in near future.