

Appendices

The computer codes written in Fortran and C++ languages, both for random number generation and calculations (integral, surface, and volume), are presented here. The codes written in Windows 7 and Kubuntu Linux operating systems are listed separately.

Appendix 1: The codes developed to generate random numbers between zero and one.

1.A- Fortran code under Windows 7

```
Real rnd, a,b
integer j
a=0
b=1
open(1,file='rnd10.txt')
call random_seed()
do j=1,10000,1
rnd=rand()*(b-a)+a
write(1,'(f6.3)')rnd
end do
end
```

1.B- C++ code under Windows 7

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#include<stdlib.h>
#include<time.h>
#include<vector>
int main()
{
randomize();
long double r;
FILE *p;
p=fopen("rnd10.txt","wb");
for(int i=1;i<10001;i++)
{
r=0.001*random(1000);
fprintf(p,"%2.3Lf\n",r);
}
fclose(p);
}
```

1.C- Fortran code under Linux

```
integer :: values(1:8), k
integer, dimension(:), allocatable :: seed
real(8) :: r
open(1,file='rnd10.txt')
call date_and_time(values=values)
call random_seed(size=k)
allocate(seed(1:k))
seed(:) = values(8)
call random_seed(put=seed)
do i=1,10000
  call random_number(r)
  write(1,'(f6.3)')r
  !print*, r
end do
end
```

1.D- C++ code under Linux

```
#include<stdio.h>
#include<math.h>
#include<stdlib.h>
#include<time.h>
int main()
{
  int randomize();
  long double r;
  FILE*p;
  p=fopen("rnd10.txt","wb");
  int i;
  srand(time(NULL));
  for(i=1;i<10001;i++)
  {
    r=0.001*(random()%1000);
    fprintf(p,"%2.3LF \n",r);
  }
  fclose(p);
}
```

Appendix 2: C++ program developed under Linux to calculate the integral

```
#include<stdio.h>
#include<math.h>
#include<stdlib.h>
#include<time.h>
int main()
{
    int randomize();
    long double r;
    int N = 10000;
    long double sum;
    long double q;
    long double I;
    long double pi = 3.141592653589793238;
    FILE*p;
    p=fopen("reuslt100.txt","wb");
    int i;
    srand(time(NULL));
    for(i=1;i<10001;i++) {
        r=0.001*(random()%1000);
        q = (pi/2)*r;
        I = (pi/(2*N))*sin(q);
        sum = sum + I;
    }
    fprintf(p,"%2.9Lf\n",sum);
    fclose(p);
}
```

Appendix 3: C++ code developed under Linux to calculate the area of a rectangle.

```
#include<stdio.h>
#include<math.h>
#include<stdlib.h>
#include<time.h>
int main()
{
    FILE*p;
    p=fopen("reuslts.txt","wb");
    int randomize();
    srand(time(NULL));
    for(int j=1;j<=100;j++)
    {
        long double r1;
        long double r2;
        long double a = 0;
        long double b = 2;
        long double c = 0;
```

```

long double d = 4;
int N = 10000;
long double x;
long double y;
long double I1;
long double I2;
long double V;
long double sum;
sum = 0.;
for(int i=1;i<=10000;i++)
{
    r1=0.001*(random()%1000);
    r2=0.001*(random()%1000);

    x = a + r1*(b-a);
    I1 = 2*x;
    y = c + r2*(d-c);
    I2 = 2*y;
    V = I1*I2;
    sum = sum + V/N;
}
fprintf(p,"%2.3Lf\n",sum);
}
fclose(p);
return 0.;
}

```

Appendix 4: C++ code developed under Linux to calculate the volume of a rectangular cube

```

#include<stdio.h>
#include<math.h>
#include<stdlib.h>
#include<time.h>
int main()
{
    FILE*p;
    p=fopen("reuslts.txt","wb");
    int randomize();
    srand(time(NULL));
    for(int j=1;j<=100;j++)
    {
        long double r1;
        long double r2;
        long double r3;
        long double a = 0;
        long double b = 2;
        long double c = 0;

```

```
long double d = 4;
long double e = 0;
long double f = 5;
int N = 10000;
long double x;
long double y;
long double z;
long double I1;
long double I2;
long double I3;
long double V;
long double sum;
sum = 0.;
for(int i=1;i<=10000;i++){
    r1=0.001*(random()%1000);
    r2=0.001*(random()%1000);
    r3=0.001*(random()%1000);
    x = a + r1*(b-a);
    I1 = 2*x;
    y = c + r2*(d-c);
    I2 = 2*y;
    z = e + r3*(f-e);
    I3 = 2*z;
    V = I1*I2*I3;
    sum = sum + V/N;
}
fprintf(p, "%2.3Lf\n", sum);
}
fclose(p);
return 0.;
```