1. Monte Carlo

Using the methods Sample Mean and Hint and Miss the program compute the integral $\int 2x^2 - 3x$, the function to integrate is in a `FUNCITON`, so this program can compute others functions with really easy changes.

The limits of the integration (2,5) are in the first lines, also the program calculate the value of the function in the limits and add or less 2, that to get a good area for Hint and Miss.

a) Sample Mean

```
    SUBROUTINE SampleMean (lowl,upl)
    IMPLICIT NONE
    REAL*8 :: upl, lowl, aux, s, integrate
    REAL*8, DIMENSION(:), ALLOCATABLE :: x
    INTEGER*8 :: i, j, k, d

    DO k = 1, 8
        d = 10*k
        ALLOCATE(x(d))
        CALL RANDOM_NUMBER(x)
        !that random numbers are in range (0,1)
        DO i = 1, d
            !new range (0,{up-low})
            x(i) = x(i)*(upl-lowl)
            !new range (lowl,upl)
            x(i) = x(i)+lowl
        ENDDO

        s = 0.
        DO i = 1, d
            s = s + f(x(i))
        ENDDO
        integrate = (upl-lowl)*s/REAL(d)
        WRITE(*,FMT='(A8,I1,A17,F9.6)') "with 10^", k, " random numbers: ",integrate
        DEALLOCATE(x)
    ENDDO
    ENDSUBROUTINE SampleMean
```

The `SUBROUTINE` is a loop from 1 to 8, exponent of 10, the number of random numbers that we will get. To save all the values is necessary `ALLOCATE` x and `DEALLOCATE` every step of the loop.

The pseudo random numbers in FORTRAN are in the range $[0, 1]$, to change the limits we multiply all the numbers by the difference between the integration limits, so the random numbers are now in the range $[0, b - a]$, finally add the inferior limit to get $[a, b]$.

To compute the integral value we sum all the values of the function valuated with the random numbers. Also write the value with the number of random numbers.

b) Hint and Miss

```
    SUBROUTINE HintAndMiss(lowl,upl,lowy,upy)
    IMPLICIT NONE
    INTEGER*8 :: nin, i, j, k, d
    REAL*8, DIMENSION(:), ALLOCATABLE :: x, y, fun
    REAL*8 :: Ar, integrate, lowl, upl, lowy, upy

    DO k = 1, 8
        d = 10**k
```

```
9           ALLOCATE(x(d),y(d),fun(d))
10          CALL RANDOM_NUMBER(x)
11          CALL RANDOM_NUMBER(y)
12          DO i = 1, d
13              x(i) = x(i)*(upl-lowl)
14              x(i) = x(i)+lowl
15
16              y(i) = y(i)*(upy-lowy)
17              y(i) = y(i)+lowy
18          ENDDO
19
20          DO i = 1, d
21              fun(i) = f(x(i))
22          ENDDO
23
24          nin = 0
25          DO i = 1, d
26              IF ( y(i) < fun(i) ) THEN
27                  nin = nin + 1
28              ENDIF
29          ENDDO
30
31          Ar = (upl-lowl)*(upy-lowy)
32          integrate = Ar*REAL(nin)/REAL(d)
33          WRITE(*,FMT='(A8,I1,A17,F9.6)') "with 10^", k, " random numbers: ",integrate
34          DEALLOCATE(x,y,fun)
35      ENDDO
36      ENDSUBROUTINE HintAndMiss
```

In this `SUBROUTINE` we have a loop from 1 to 8 as the previous one. But additionally we call more random numbers in `y` changing the range as equal to `x`.

Then compute the value of the function with every random `x`, and compare the value of random `y` and the function with random `x`. If the value is less we consider that the point is under the function. After the comparison in the lines 25 to 29 we can compute integral as the fraction of the rectangle area and how many points are under the function and all points.

c) Discussion

After run the program we can see Sample Mean is faster, whereas we get more disperse values than Hint and Miss that is near to the exact value as we increase the number of random numbers. However, the average of all Sample Mean is close to the exact value, consider the time cost.

Random numbers	Sample Mean	Hint and Miss
10^1	58.396840	55.500000
10^2	56.799374	56.610000
10^3	46.279869	47.952000
10^4	45.368156	46.620000
10^5	46.310320	46.585590
10^6	53.178851	46.513107
10^7	45.301157	46.495480
10^8	48.386415	46.510594
average	50.002623	

Since we are using random numbers, the values in the tables will change. We know the exact value 46.5, since the integral has a analytic solution.