

1.

N = 10

Pi = 4.000000

Pi = 4.000000

Pi = 2.400000

Pi = 2.400000

Pi = 3.200000

Pi = 3.200000

Pi = 3.600000

Pi = 3.200000

Pi = 2.800000

Pi = 2.800000

Mean = 3. 3.060000

Standard Deviation = 0.620000

%%%%%%%%%

N = 100

Pi = 3.000000

Pi = 3.400000

Pi = 3.360000

Pi = 3.080000

Pi = 3.000000

Pi = 3.200000

Pi = 3.200000

Pi = 2.920000

Pi = 3.240000

Pi = 3.360000

Mean = 3.176000

Standard Deviation = 0.161196

%%%%%%%%%

N = 1000

Pi = 3.120000

Pi = 3.052000

Pi = 3.136000

Pi = 3.140000

Pi = 3.196000

Pi = 3.168000

Pi = 3.212000  
Pi = 3.164000  
Pi = 3.136000  
Pi = 3.176000  
Mean = 3.150000  
Standard Deviation = 0.042605

%%

N = 100000

Pi = 3.132120  
Pi = 3.143960  
Pi = 3.143440  
Pi = 3.148280  
Pi = 3.134720  
Pi = 3.141800  
Pi = 3.140960  
Pi = 3.145720  
Pi = 3.143200  
Pi = 3.144920  
Mean = 3.141912  
Standard Deviation = 0.004699

%%

N = 1000000

Pi = 3.142244  
Pi = 3.140948  
Pi = 3.140520  
Pi = 3.141608  
Pi = 3.142768  
Pi = 3.141052  
Pi = 3.141264  
Pi = 3.142072  
Pi = 3.135720  
Pi = 3.140436  
Mean = 3.140863  
Standard Deviation = 0.001858

%%

N = 10000000

Pi = 3.141286  
Pi = 3.140210  
Pi = 3.141008  
Pi = 3.141914  
Pi = 3.141324  
Pi = 3.141626  
Pi = 3.141693  
Pi = 3.141966  
Pi = 3.142298  
Pi = 3.141510  
Mean = 3.141484  
Standard Deviation = 0.000553

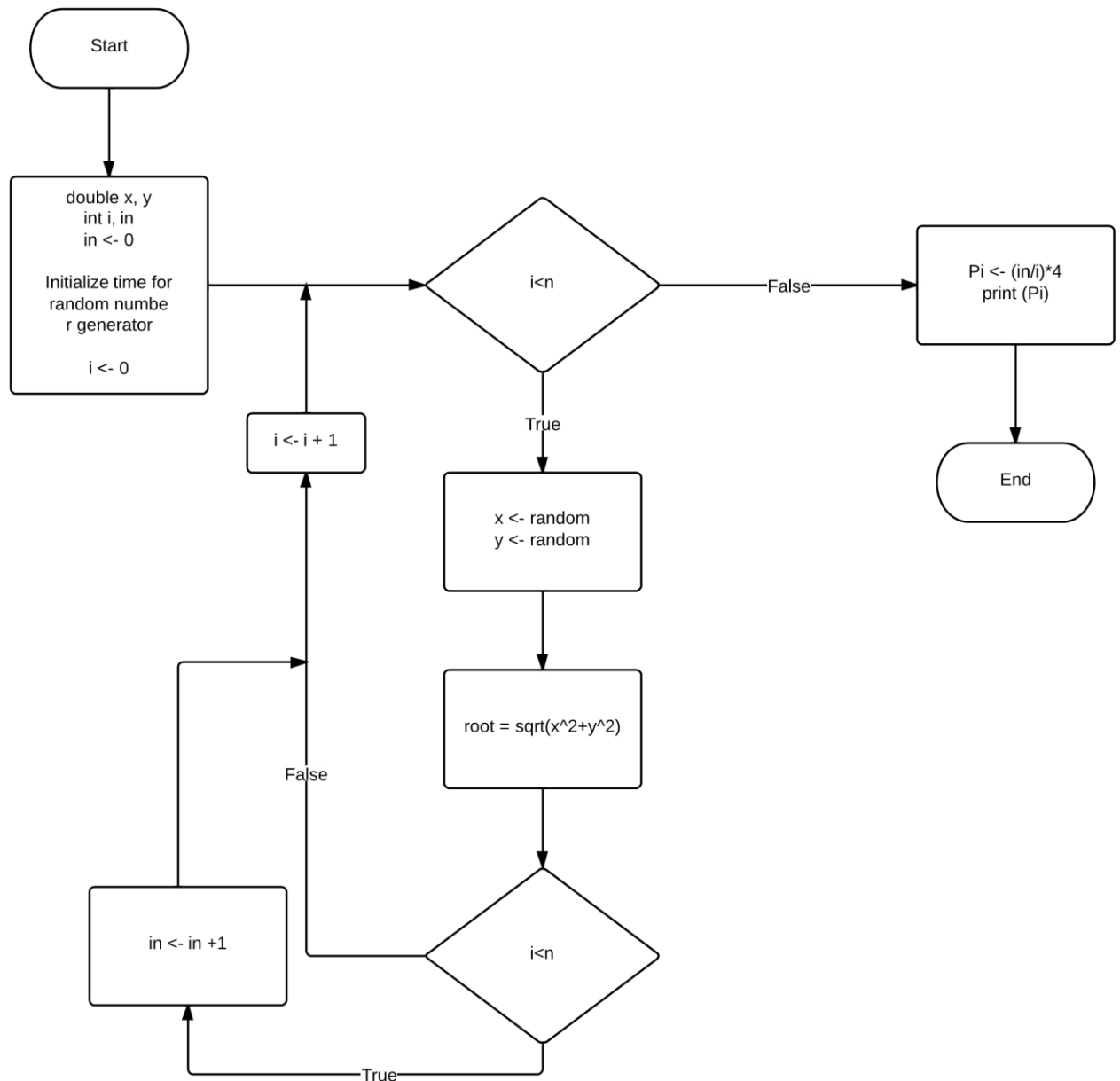
%%

N = 100000000

Pi = 3.141269  
Pi = 3.141788  
Pi = 3.141474  
Pi = 3.141617  
Pi = 3.141602  
Pi = 3.141610  
Pi = 3.141564  
Pi = 3.141331  
Pi = 3.141776  
Pi = 3.141775  
Mean = 3.141581  
Standard Deviation = 0.000171

%%

As the N value increases, the Values of Pi become more accurate, the mean converges on the actual Pi value, and the standard deviation value decreases.



2.

**obelix[64]% gcc prog2.c**

**obelix[65]% a.out**

**Enter n value:100**

**Enter n value:133**

**Enter n value:60**

**Enter n value:0**

**Enter n value:3**

8    1    6

3    5    7

4    9    2

<-- These show that the following values  
are incorrect input values.

The user is prompted again for  
a value.

<-- Correct input value

%%

**obelix[67]% a.out**

**Enter n value:1**

1

%%

**obelix[68]% a.out**

**Enter n value:7**

30	39	48	1	10	19	28
38	47	7	9	18	27	29
46	6	8	17	26	35	37
5	14	16	25	34	36	45
13	15	24	33	42	44	4
21	23	32	41	43	3	12
22	31	40	49	2	11	20

%%

**obelix[68]% a.out**

**Enter n value:5**

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9





obelix[32]% a.out

Enter the amount in dollars: 30

The smallest number of \$20-\$1 bills/coins necessary to pay the dollar amount: 30

1 20\$, 1 10\$, 0 5\$, 0 2\$, 0 1\$

%%%

obelix[33]% a.out

Enter the amount in dollars: 524132

The smallest number of \$20-\$1 bills/coins necessary to pay the dollar amount: 524132

26206 20\$, 1 10\$, 0 5\$, 1 2\$, 0 1\$

%%%

obelix[34]% a.out

Enter the amount in dollars: 765

The smallest number of \$20-\$1 bills/coins necessary to pay the dollar amount: 765

38 20\$, 0 10\$, 1 5\$, 0 2\$, 0 1\$

