

```
//Calculation of PI-Archimedes Method
```

```
//Author: Christoher Rayappan
```

```
/* Algorithm
```

To calculate the perimeter of the circle with a diameter of one unit, which happens to be equal to PI

Inscribe a circle with polygon A . The perimeter of polygon A gives the Lower bound of PI. Circumscribe the same circle with polygon B.

The perimeter of the second polygon gives the upper bound of PI.

Increase the sides of polygon A and polygon B and calculate the perimeter by Geometric mean and Harmonic mean. This gives a closer approximation to PI*/

```
#include<stdio.h>
```

```
#include<math.h>
```

```
main()
```

```
{
```

```
int N, x1;
```

```
int i=0;
```

```
int j=1;
```

```
double p[6144];
```

```
double P[6144];
```

```
printf("\nEnter the N value for 6*(2^N) sided polygons\n");
```

```
scanf("%d",&N);
```

```
while(N>10)
```

```
{
```

```
    printf("\n Enter a lower N value for 6*(2^N) sided polygons\n");
```

```
    scanf("%d",&N);
```

```

}
x1=6*pow(2,N);
printf("\n-----\n");
printf("\nCalculation of PI value for %d sided polygons\n",x1);
printf("\n-----\n");
if(x1==6)
{
    printf("\nLower bound of PI is 3\n");
    printf("\nUpper bound of PI is 3.4641\n");
    return 0;
}
else
    p[i] = 3; //Lower bound PI value for 6 sided polygon
    P[i] = 3.4641; // Upper bound PI value for 6 sided polygon
for(i= 0; i<N;i++)
    {
        P[i+1]= (2*p[i]*P[i])/(p[i]+P[i]); //Upper bound calculated by Harmonic mean
        p[i+1]= sqrt(p[i]*P[i+1]); //Lower bound calculated by Geometric mean
    }
    printf("\nLower bound of PI is %f\n",p[i]);
    printf("\nUpper bound of PI is %f\n",P[i]);
return 0;
}

```