[ ]:
```
h = 0.5
a = 1
b = 4

steps = Int(ceil((b - a) / h)) + 1
X_vec = a:h:b
Y_vec = zeros(steps)

for i in 1:steps
    Y_vec[i] = acos(log2(X_vec[i]/2))
end

print(" x  |  acos(log2(x/2))\n")
print("----------------------\n")
for i in 1:steps
    print(X_vec[i], " | ", Y_vec[i], "\n")
end
```

```
 x  |  acos(log2(x/2))
----------------------
1.0 | 3.141592653589793
1.5 | 1.9987803462433056
2.0 | 1.5707963267948966
2.5 | 1.2430310324200673
3.0 | 0.9459625046862551
3.5 | 0.6311407394425016
4.0 | 0.0
```

[ ]:
```
x = 2.2
mid = X_vec[Int(ceil((steps/2))) + 1]
print(" Our x: ", x, "\n")
print("middle: ", mid, "\n")
print("Count knots: ", length(X_vec), "\n")
```

```
 Our x: 2.2
middle: 3.0
Count knots: 7
```

Так как число узлов нечетно, но при этом значение x не выполняет неравенство: a <= x < a + h/4, то многочлен Стирлинга не подходит.

Уберем последний узел интерполяции (4, 0), тогда кол-во узлов станет четным, и можно будет использовать многочлен Бесселя, тогда серединой окажется узел (2, 1.5708)

[ ]:
```
steps -= 1
X_vec = X_vec[1:steps]
Y_vec = Y_vec[1:steps]
b -= h
```

```
print(" x  |  acos(log2(x/2))\n")
print("-----------------------\n")
for i in 1:steps
    print(X_vec[i], " | ", Y_vec[i], "\n")
end

print("\n")

mid = X_vec[Int(ceil(steps/2))]
print(" Our x: ", x, "\n")
print("middle: ", mid, "\n")
print("Count knots: ", length(X_vec), "\n")
```

```
 x  |  acos(log2(x/2))
-----------------------
1.0 | 3.141592653589793
1.5 | 1.9987803462433056
2.0 | 1.5707963267948966
2.5 | 1.2430310324200673
3.0 | 0.9459625046862551
3.5 | 0.6311407394425016

 Our x: 2.2
middle: 2.0
Count knots: 6
```

Теперь все условия для интерполяционного члена Бесселя выполняются.

[ ]:
```
differences = zeros(steps, steps)

for i in 1:steps
    differences[i, 1] = Y_vec[i]
end

for i in 2:steps
    for j in 1:(steps-i+1)
        differences[j, i] = differences[j+1, i-1] - differences[j,␣
  ↪i-1]
    end
end

print("Конечные разности: \n\n")
for i in 1:steps
    for j in 1:steps
        print(round(differences[i, j], digits=3), "\t")
    end
    print("\n")
```

```
end
```

Конечные разности:

```
3.142   -1.143  0.715   -0.615  0.545   -0.524
1.999   -0.428  0.1     -0.07   0.021   0.0
1.571   -0.328  0.031   -0.048  0.0     0.0
1.243   -0.297  -0.018  0.0     0.0     0.0
0.946   -0.315  0.0     0.0     0.0     0.0
0.631   0.0     0.0     0.0     0.0     0.0
```

[ ]:
```
function BesselPoly(X_vec, diffs, x, h)
    size = length(X_vec)
    mid_ind = Int(size / 2)
    q = (x - X_vec[mid_ind]) / h

    polynom = 1.0
    answer = (diffs[mid_ind, 1] + diffs[mid_ind + 1, 1]) / 2
    answer +=(q - 0.5) * diffs[mid_ind, 2]

    for n in 3:2:size
        polynom *= (q + (n - 3) / 2) * (q - (n - 1) / 2)

        answer += polynom / factorial(n - 1) * (diffs[Int(mid_ind -␣
  ↪(n - 1) / 2), n] + diffs[Int(mid_ind - (n - 3) / 2), n]) / 2
        answer += (q - 0.5) * polynom / factorial(n) *␣
  ↪diffs[Int(mid_ind - (n - 1) / 2), n + 1]
    end

    return answer
end


print("           f(x) = ", BesselPoly(X_vec, differences, x, h),␣
  ↪"\n")
print("acos(log2(x/2)) = ", acos(log2(x/2)), "\n")
```

```
           f(x) = 1.4381329377767518
acos(log2(x/2)) = 1.4328557728353513
```