

Mateusz Sławiński 222B

Sprawozdanie - lab2

1) Napisz sparametryzowaną funkcję wyznaczającą sygnał tonu prostego $s(t)$ w dziedzinie liczb rzeczywistych. Wygeneruj wykres dla $t \in \langle 0; \hat{A} \rangle$, jako parametry inicjalizujące przyjmij: $A = 1.0$ [V], $f = \hat{B}$ [Hz], $\varphi = \hat{C} \cdot \pi$ [rad], częstotliwość próbkowania f_s dopasuj tak aby wykres sygnału był czytelny.

Będzie to już spróbkowany lecz jeszcze nie skwantyzowany sygnał.

```
int a = 9, b = 9, c = 4;

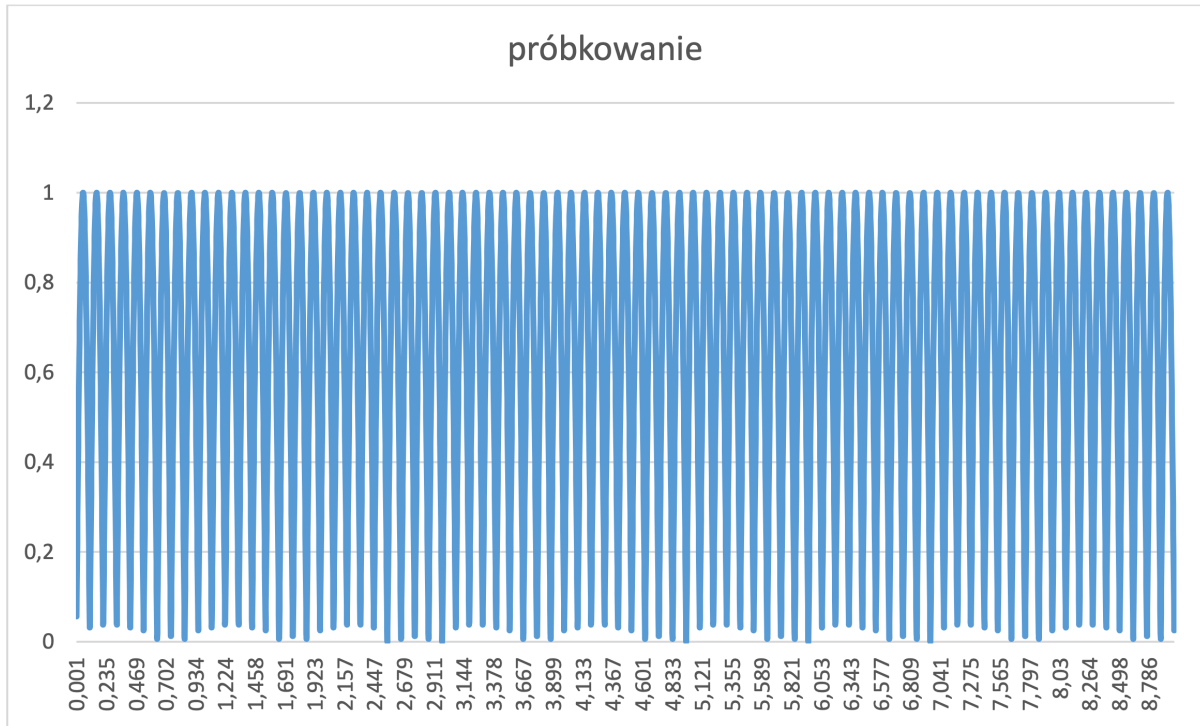
float funkcja( double czas, double amplituda, double
czestotliwosc, double przesuniecie )
{
    return amplituda * sin( 2 * M_PI * czestotliwosc * czas +
przesuniecie );
}

int lab2() {
    ofstream plik;

    float poczatek1 = 0;
    float krok1 = 0.05;
    int n = 0;
    float t = poczatek1 + (n * krok1);

    //1
    plik.open("probkowanie.txt");
    while (t <= 9)
    {
        plik << t << " " << funkcja(t, 1.0, 9, c * M_PI) <<
"\n";
        n++;
        t = poczatek1 + (n * krok1);
    }
    plik.close();
    return 0;
}
```

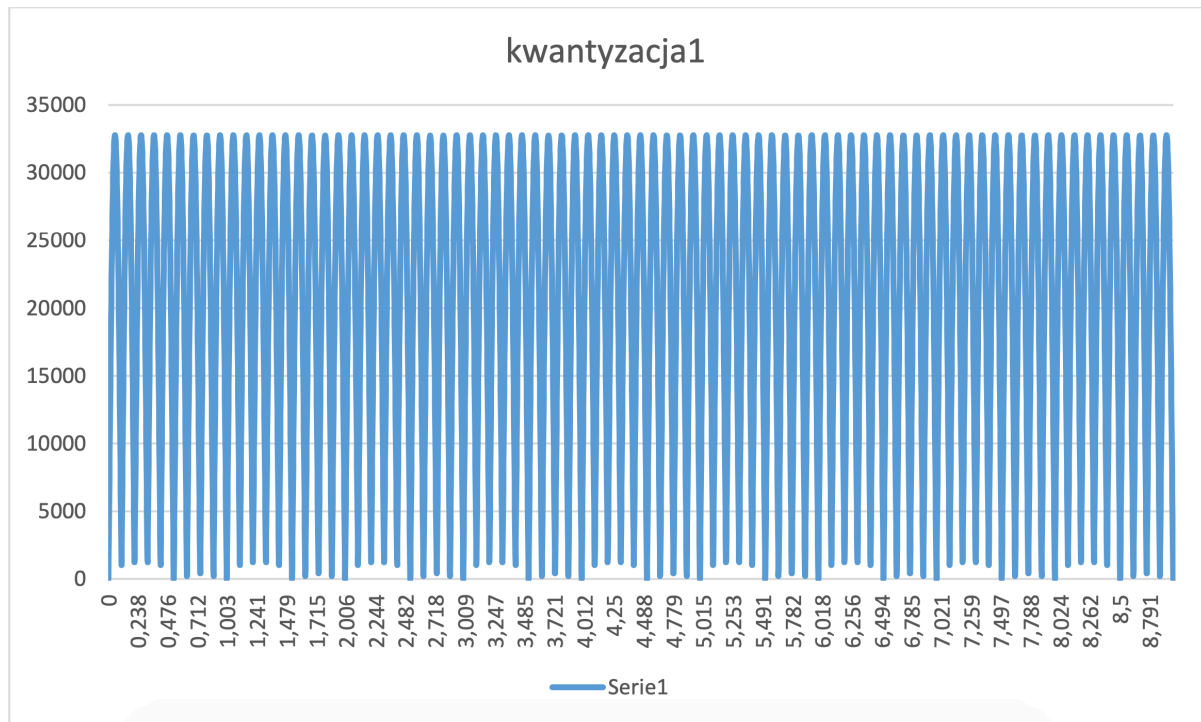
WYKRES próbkowanie:



2) Napisz funkcję kwantyzującą sygnał z zadania pierwszego do zadanej rozdzielczości kwantyzacji $q = 16 \rightarrow 2^q$. Wygeneruj wykres skwantyzowanego sygnału.

```
float kwantyzacja( double signal, int q ) {  
    return ceil( ( pow( 2, q ) ) * (signal / 2) );  
}  
  
//2  
int lab2() {  
    ofstream plik;  
    n = 0;  
    t = poczatek1 + (n * krok1);  
    plik.open("kwantyzacja1.txt");  
    while (t <= 9)  
    {  
        float sygnal = funkcja(t, 1.0, 9, c * M_PI);  
        plik << t << " " << kwantyzacja( sygnal, 16 ) <<  
            "\n";  
        n++;  
        t = poczatek1 + (n * krok1);  
    }  
    plik.close();  
    return 0;  
}
```

WYKRES kwantyzacja, q=8:



3) Wygeneruj wykres sygnału z zadania drugiego zmniejszając o połowę f_s i q .

```
float kwantyzacja( double signal, int q ) {
    return ceil( ( pow( 2, q ) ) * (signal / 2) );
}

//3
int lab2() {
    ofstream plik;
    n = 0;
    t = poczatek1 + (n * krok1);
    plik.open("kwantyzacja2.txt");
    while (t <= 9)
    {
        float sygnal = funkcja(t, 1.0, 9 * 2, c * M_PI);
        plik << t << " " << kwantyzacja( sygnal, 8 ) << "\n";
        n++;
        t = poczatek1 + (n * krok1);
    }
    plik.close();
    return 0;
}
```

WYKRES kwantyzacja, q=16:

