20-AMALIY MASHG'ULOT. STRING SINFI. STRING TURIDAGI SATRLARDAN FOYDALANIB DASTURLAR TUZISH

NAZARIY QISM

STRING TURIDAGI SATRLAR BILAN ISHLASH

7.1. String turidagi satrlarni e'lon qiling va ularni chop eting.

```
	#include <iostream>
#include <string>
using namespace std;

void main()
{
	string A = "Erali"; //1-usul
	string B(A); //2-usul
	string C("Salom, Dunyo!"); //3-usul
	string S = C; //4-usul

	cout << A << endl; //Chop etish //1-usul
	cout << C.at(1) << endl; //2-usul 1-element ekranga chop etiladi


	return;
}
```

7.2. Bir satrni ikkinchi satrga nusxalang.

```
#include <iostream>
#include <string>
using namespace std;

void main()
{
	string A = "Erali";
	string B;
	string C("Salom, Dunyo!");
	string S;

	B.assign(A); //A ni B ga nusxalash B=A ga ekvivalent
	S.assign(C, 0, 6); //C satrning 0-o'rnidan 6 ta belgisini nusxalaydi
	S += A; //A ni S ga qo'shadi
	cout << S << endl;
	return;
}
```

7.3. Quyidagicha satrlar berilgan: "Klavikomp" va "aturayuter". Ushbu satrlardan kompyuter va klaviatura so'zini hosil qiling.

```
	#include <iostream>
#include <string>
```

```
using namespace std;

void main()
{
	string A = "Klavikomp";
	string B = "aturayuter";
	A.insert(5, B, 0, 5); //A satrning 5-belgisidan keyin B satrning 0-belgisidan 5 ta
belgisini joylashitiradi natijada Klaviturakomp so'zi hosil bo'ladi
	A.erase(10, 4); //10-belgisidan boshlab 4 tasini o'chiradi, klaviatura so’zi hosil
bo’ladi
	cout << A << endl;

	return;
}
```

7.4. Bizga “Paxtakor”, “Bunyod” kabi ikkita satr berilgan. “Bunyodkor” satrini hosil qiling.

```
#include <iostream>
#include <string>
using namespace std;

void main()
{
	string A = "Bunyod";
	string B = "Paxtakor";
	string C;
	C = B.substr(5, 3); //kor ni ajratib olish;
	A.append(C, 0, 3); //A+=C ga ekvivalent;
	cout << A << endl;

	return;
}
```

7.5. Bizga string turidagi satr berilgan. Uning har bir belgisini alohida-alohida ekranga chiqaring.

```
#include <iostream>
#include <string>
#include <cstdlib>
using namespace std;

void main()
{
	int n;
	string A = "Matn berilgan uni alohida ekranga chiqarish kerak";
	//1-usul
	for (int i = 1; i <= 20; i++)
		cout << A.at(i) << endl;

	return;
}
```

7.6. Satr berilgan. Uning quyidagi xossalarini aniqlang: satr uzunligini, satr egallagan xotira hajmini, satr o'lchami, satrnig bo'shligini

```
#include <iostream>
#include <string>
#include <cstdlib>
using namespace std;

void main()
{
	int n, n1, n2, n3;
	string A = "Satr berilgan";

	n = A.size(); //satr o'lchami
	n1 = A.length(); //satr uzunligi
	n2 = A.max_size(); //Maksimal uzunlik
	n3 = A.capacity(); //Satr egallagan xotira hajmi
	if (A.empty() == true) //Agar satr bo'sh bo'lsa
		cout << "Satr bo'sh";
	else
		cout << "Satr bo'sh emas";
	cout << n;
	cout << n1;
	cout << n2;
	cout << n3;
	return;
}
```

7.7. Familiya, ismi shariflari bilan talabalar ro'yxati berilgan. Ro'yxat alvafit bo'yicha tartiblansin.

```
#include <iostream>
#include <malloc.h>
using namespace std;
void main()
{
	const int FISH_uzunligi = 50;
	string *Talaba;
	char *Satr = (char*)malloc(FISH_uzunligi);
	unsigned int talabalar_soni;
	char son[3];
	do
	{
		cout << "Talabalar sonini kiriting:"; cin >> son;
	} while (talabalar_soni = atoi(son) <= 0);
	Talaba = new string[talabalar_soni];
	cin.ignore();
	for (int i = 0; i < talabalar_soni; i++)
	{
		cout << i + 1 << "-talabaning Familiyasi Ismi va SHarifi:";
		cin.getline(Satr, 50);
		Talaba[i].assign(Satr);
	}
	bool almashdi = true;
```

```
	for (int i = 0; i < talabalar_soni - 1 && almashdi; i++)
	{
		almashdi = false;
		for (int j = i; j < talabalar_soni - 1; j++)
		if (Talaba[j].compare(Talaba[j + 1]>0))
		{
			almashdi = true;
			strcpy(Satr, Talaba[j].data());
			Talaba[j].assign(Talaba[j + 1]);
			Talaba[j + 1].assign(Satr);
		}
	}
	cout << "Alfavit bo'yich atartiblangan ro'yxat:\n";
	for (int i = 0; i < talabalar_soni; i++)
		cout << Talaba[i] << endl;
	delete[] Talaba;
	free(Satr);
	return;
}
```

7.8. Berilgan matnda nechta so’z bor bo’lsa hammasini alohida ustun shaklida chop eting.

```
#include <iostream>
#include <string.h>
using namespace std;
int main()
{
	char S[] = "Erali Axror dunyo olam baxt g'urur o'lka";

	for (int i = 0; i < strlen(S); i++)
	{
		if (S[i] != ' ')
			cout << S[i];
		else
			cout << endl;
	}

}
```

7.9. b harfi bilan boshlanuvchi satr nechtaligini aniqlang.

```
#include <iostream>
#include <string>

using namespace std;
int main()
{
	int satr_u, k = 0;
	char S[200];
	cout << "Matnni kiriting" << endl;
	cin.getline(S, 200); //satrn kiritish uchun 200 ta belgi
	satr_u = strlen(S); //satrning uzunligini aniqlash
```

```
	if (S[0] == 'b' || S[0] == 'B') //Birinchi belgini tekshrish
		k = 1;
	for (int i = 0; i <= satr_u; i++)
	if (S[i] == ' ' && ((S[i + 1] == 'B') || (S[i + 1] == 'b'))) //Agar O'zidan oldin
bitta probel bo'lsa va b yoki B harflari bilan boshlansa
		k++;
	cout << "Matndagi b harfi bilan boshlanuvchi so'zlar soni: " << k << endl;


	return 0;
}
```

7.10. Satr berilgan. Undan biror belgining nechtaligini hisoblab chiqing.

```
#include <iostream>
#include <string>

using namespace std;
int main()
{
	int satr_u, s = 0;
	char S[200];
	char k;
	cout << "Matni kiriting" << endl;
	cin.getline(S, 200); //satrn kiritish uchun 200 ta belgi
	satr_u = strlen(S); //satrning uzunligini aniqlash
	cout << "bu matnda qaysi belgi nechtaligini izlayapsiz?" << endl;
	cin >> k;
	for (int i = 0; i <= satr_u; i++)
	if (S[i] == (char)k) //Agar k belgisiga teng bo'lsa. Bu yerda turga keltirish
amali qo'llanilmoqda
		s++;
	cout << "Matnda " << k << " belgisi " << s << " marta uchraydi" << endl;


	return 0;
}
```

7.11. Matn berilgan. Matnda eng qisqa so’z va eng uzun so’z uzunliklarini toping.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
	int satr_u, s = 0, max = 0;
	char S[200];
	cout << "Matnni kiriting" << endl;
	cin.getline(S, 200);
	satr_u = strlen(S);
	for (int j = 0; j <= satr_u; j++)
	{
		if (S[j] != ' ')
			s++;
		else
```

```
		{
			if (s>max)
				max = s;
			s = 0;
		}
	}
	cout << "Bu matndagi eng uzun so'z" << max << "ta harfdan tashkil topgan" << endl;
	return 0;
}
```

7.12. Orasida ikki nuqta bo'lgan satr berilgan. Ungacha bo'lgan belgilar qanchaligini aniqlang.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
	int satr_u, s = 0;
	char S[200];
	cout << "Matnni kiriting" << endl;
	cin.getline(S, 200);
	satr_u = strlen(S);
	for (int j = 0; j <= satr_u; j++)
	{
		if (S[j] != ':')
			s++;
		else
		{
			cout << "Bu matndagi ikki nuqtagcha belgilar soni:" << s << " ta" <<
endl;
			break;
		}
	}

	return 0;
}
```

7.13. Nuqta bilan tugaydigan satr berilgan. Uchta harfdan iborat so'zlarni chiqaring.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
	int satr_u, s = 0;
	char S[200];
	cout << "Matnni kiriting" << endl;
	cin.getline(S, 200);
	satr_u = strlen(S);
	cout << "Bu matndagi uchta harfdan tashkil topgan so'zlar:" << endl;
	for (int j = 0; j <= satr_u; j++)
	{
		if (S[j] != ' ')
```

```
                s++;
            else
            {
                if (s == 3)
                    cout << S[j - 3] << S[j - 2] << S[j - 1] << endl;
                s = 0;
            }
        }

        return 0;
}
```

7.14. Satr berilgan. Unda ‘abc’ qismiy satr necha marta uchrashini aniqlang.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
        int satr_u, s = 0;
        char S[200];
        cout << "Matnni kiriting" << endl;
        cin.getline(S, 200);
        satr_u = strlen(S);
        cout << "Bu matndaabc qismiy satr:" << endl;
        for (int j = 2; j <= satr_u; j++)
        {
            if (S[j - 2] == 'a' && (S[j - 1] = 'b' && (S[j] == 'c')))
                s++;
        }
        cout << s << " marta uchraydi" << endl;
        return 0;
}
```

7.15. Orasida bitta ochilgan qavs va bitta yopilgan qavs mavjud bo’lgan belgili satr berilgan. Bu qavslar barcha belgilarni ekranga chiqaring.

```
#include <iostream>
#include <string.h>
using namespace std;
int main()
{
        char S[200];
        char belgi[] = "()";
        char* Satr;
        cin.getline(S, 200);
        Satr = strtok(S, belgi); //S satrdan belgi satrlarini olib tashlaydi
        if (Satr) cout << Satr << endl;
        while (Satr)
        {
            Satr = strtok(NULL, belgi); //
            if (Satr) cout << Satr << endl;
        }
        return 0;
```

```
}
```

7.16. a harfi bilan tugovchi so’zlar sonini aniqlang.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
	int satr_u, s = 0;
	char S[200];
	cout << "Matnni kiriting" << endl;
	cin.getline(S, 200);
	satr_u = strlen(S);
	cout << "Bu matnda a harfi bilan tugovchi so'zlar:" << endl;
	for (int j = 0; j <= satr_u; j++)
	{
		if (S[j] == 'a' && (S[j + 1] == ' '))
			s++;
	}
	cout << s << " marta uchraydi" << endl;
	return 0;
}
```

7.17. Satr berilgan. Boshlanishi va tugashi bir xil harfdan iborat bo’lgan so’zni toping.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
	int satr_u, s = 0, i = 0, j = 1;
	char S[200];
	cout << "Matnni kiriting" << endl;
	cin.getline(S, 200);
	satr_u = strlen(S);
	cout << "Bu matnda boshlanishi va oxiri bir xil harf bilan tugovchi so'zlar soni:"
<< endl;
	i = 0;
	while (i <= satr_u)
	{
		if (S[i] == S[j] && S[j + 1] == ' ')
		{
			s++;
			i++;
		}
		else
			j++;
	}
	cout << s << " marta uchraydi" << endl;
	return 0;
```

```
}
```

7.18. Berilgan so’z satrda necha marta uchrashini aniqlang.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{

	char S[200], S1[20];
	char *k = 0;
	cout << "Matnni kiriting" << endl;
	cin.getline(S, 200);
	cout << "Matndan qaysi so'zni izlayapsiz?" << endl;
	cin.getline(S1, 15);
	k = strstr(S, S1); // S satrga S1 satrning kirishini aniqlaydi
	if (k != 0)
		cout << "Bu so'z mavjud" << endl;
	else
		cout << "Bu so'z mavjud emas" << endl;
	return 0;
}
```

7.19. Satr bitta so’zdan iborat. Uni chapdan o’ngga va o’ngdan chapga qarab o’qilganda bir xil bo’lishini tekshiring.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{

	char S[20], *S1[20];
	char *k;

	cout << "So'zni kiriting" << endl;
	cin.getline(S, 20);
	cout << "So'zning teskarisi" << endl;
	*S1 = strrev(S); //so'zning teskarisi
	cout << *S1 << endl;
	k = strstr(S, *S1);
	if (k != 0)
		cout << "Bu so'z polindrom";
	else
		cout << "Bu so'z polindrpom emas";
	return 0;
}
```

7.20. Berilgan satrda ochilgan qavslar soni yopilgan qavslar soni bilan bir xilligini tekshiring.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{

	char S[100];
	int k = 0, l = 0;
	//-----------Matnni kiritish jarayoni --------------//

	cout << "Qavslardan iborat matn kiriting:" << endl;
	cin.getline(S, sizeof(S));

	//-------Matndagi ochilgan qavslar sonini tekshirish -------------//

	for (int i = 1; i <= sizeof(S); i++)
	{
		if (S[i] == '(')
			k++;
		if (S[i] == ')')
			l++;
	}
	if (k == l)
		cout << "Ochilgan qavslar soni yopilgan qavslar soniga teng!";
	else
		cout << "Teng emas!";
	return 0;
}
```

7.21. Ikkita A va B satr berilgan. A satrdagi harflardan B satrni tuzish mumkinligini (harflarni o'rnini ishlatish mumkin, lekin ularni bir martadan ortiq ishlatish mumkin emas) tekshiradigan dastur tuzing. Masalan: A: INTEGRAL, B:AGENT – tuzish mumkin, B:AGENT – tuzish mumkin emas.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
	int const n = 10, m = 5;
	char A[n], B[m];
	int S = 0;

	//-----------So'zlarni kiritish --------------//

	cout << "Ikkita matnni kiriting:" << endl;
	cout << "A="; cin.getline(A, sizeof(A));
	cout << "B="; cin.getline(B, sizeof(B));

	//--------------B so'zning harfalri A so'zida mavjudligini tekshirish-------------
--//
```

```
	for (int i = 1; i <= strlen(B); i++)
	{
		for (int j = 1; j <= strlen(A); j++)
		if (B[i] == A[j])
		{
			S++;
			break;
		}
	}
	//Agar S B ning uzunligigateng bo'lsa mumkin aks holda mumkin emas
	if (S == strlen(B))
		cout << "Mumkin";
	else
		cout << "Mumkin emas";
	return 0;
}
```

7.22. Berilgan belgilar satridan belgilarni shunday tanlangki, ular shu satrda bir marta uchrasin, hamda ularning matndagi tartibi o'zgarmasin.

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
	int const n = 30;
	char A[n];
	int S = 0;

	//-----------Matnni kiritish --------------//

	cout << "Matnni kiriting:" << endl;
	cout << "A="; cin.getline(A, sizeof(A));

	//--------------A so'zida faqat bir marta uchragan harflarni ekranga chop etish --
-------------//
	for (int i = 0; i <= strlen(A); i++)
	{
		for (int j = 0; j <= strlen(A); j++)
		if (A[i] == A[j])
			S++;
		if (S == 1)
			cout << A[i] << " ";
		S = 0;
	}
	cout << endl;
	return 0;
}
```

7.23. Ikkilik sanoq sistemasidagi son berilgan. Bu son to'g'ri kiritilganligini tekshiring (uning yozuvida 0 va 1 raqamlari bo'lishi kerak). Agar son noto'g'ri kiritilgan bo'lsa, kiritish takrorlansin. Son to'g'ri kiritilgandan keyin uni o'nlik sanoq sitemasiga o'tkazing

```
#include <iostream>
#include <string>
#include <stdlib.h>
using namespace std;
int main()
{
	int const n = 10;
	char A[n], k1[n];
	int k = 0;
	long a;


	//-----------Kiritilgan sonni ikkilik SSga tegishli ekanligini tekshirish --------
------//

	while (k == 0)
	{
		cout << "Sonni kiriting" << endl;
		cin.getline(A, sizeof(A));
		for (int i = 0; i <= strlen(A) - 1; i++)
		{
			if (A[i] == '1' || A[i] == '0')
				k = 1;
			else
			{
				k = 0;
				break;
			}
		}
		cout << "k=" << k << endl;
	}
	//------------Berilgan son ko'rinishidagi matnni tipini o'zgartirish-------------
	a = atol(A); //Matnni songa aylantirish

	//------Ikkilik SS dan o'nlik SS ga o'tish---------//
	cout << "k1=" << k1;

	return 0;
}
```

7.24. Berilgan satrdagi barcha ortiqcha bo'sh joy belgilarini o'chiring.

```
#include <iostream>
#include <string>
#include <stdlib.h>
using namespace std;
int main()
{
	int const n = 100;
```

```
	char A[n];

	cout << "Matnni kiriting:" << endl;
	cin.getline(A, sizeof(A));
	for (int i = 0; i <= strlen(A) - 1; i++)
	if (A[i] == ' ')
		A[i] = A[i + 1];

	cout << "Yangi matn:";
	cout << A;


	return 0;
}
```

7.25. Berilgan matn uchun harflarda faqrli bo'lgan ketma-ket belgilarning maksimal uzunligini aniqlang.

```
#include <iostream>
#include <string>
#include <stdlib.h>
using namespace std;
int main()
{
	int const n = 100;
	char A[n];
	int S = 0, k = 0, max = 0;

	cout << "Matnni kiriting:" << endl;
	cin.getline(A, sizeof(A));
	for (int i = k; i <= strlen(A) - 1; i++)
	{
		for (int j = i; j <= strlen(A) - 1; j++)
		if (A[j] >= 48 && A[j] <= 57) //Raqamlarni ASCII kodi 48 va 57 oralig'ida
joylashgan
			S++;
		else
		{
			if (S > max)
				max = S;
			k = j + 1;
			break;
		}
		S = 0;
	}

	cout << "Matndagi eng uzun raqamlar ketma-ketligi soni:";
	cout << max;


	return 0;
}
//Eslatma: bu masalada harflardan farqli belgi sifatida faqat son olingan
```