**15 билет**

**Рекурсия. Рекурсивті алгоритмдер**

**Өзін өзі шақыратын функция рекурсия деп аталады. Рекурсия тереңдігі дегеніміз – функция мәнін есептеуде өзін-өзі шақыру саны. Рекурсивті программалау стек принципіне сүйенеді.**

**Рекурсия түрлері мынадай:**

**сызықтық рекурсия;**

**параллель рекурсия;**

**қосалқы рекурсия;**

**жоғары ретті рекурсия.**

**Рекурсивті функциялар**

**Ары қарай қарастыру үшін бізге бірқатар анықтамалар қажет болады. Айталық, Х және Ү екі жиыны бар болсын.**

**Егер Х жиынының кейбір элементтеріне Ү жиының бірмәнді анықталған элементтері сәйкестендірілген болса, онда Х-тан Ү-ке жекелеме функция (частичная функия) берілген деп аталады. Ү-те сәйкес элементтері бар Х-тің элементтерінің жиынтығы функияның анықталу облысы деп аталады, ал Х-тің элементтеріне сәйкес келетін Ү-тің элементтерінің жиынтығы функия мәндерінің жиынтығы деп аталады. Егер функцияның Х-тен Ү-ке анықталу облысы Х жиынымен сәйкес келсе, онда функция барлық жерде анықталған деп аталады.**

**Рекурсивті функия ұғымына сүйеніп алгоритм ұғымын нақты анықтамасын құрудың бастапқы идеясы мынада жатыр: кез-келген берілгендерді (әрине дискретті) қандай-да бір санау жүйесінде натурал сандармен кодтауға болады, және сонда оларды кез-келген жолмен түрлендіру - есептеу операцияларының тізбегіне келтіріледі, ал өңдеу нәтижесі де сол сияқты бүтін санды**

**Бұл жағдайда осы сандық функцияға бірдей кез-келген алгоритм оның мәнін есептейді, ал оның элементар қадамдары кәдімгі арифметикалық және логикалық амалдар болып табылады. Мұндай функциялар есептелінетін функциялар деп аталады.**

**Рекурсивті алгоритмдер мысалдары:**

**6!- алты факториялды есептеу алгоритмі:**

**#include**

**double fact(int n);**

**int n=6;**

**double f;**

**f=fact(n);**

**printf(‘6!=%10.0f\n”,f);**

**return (0);**

**}**

**double fact(int n)**

**{**

**if (n<1) return(1.0)**

**else**

**return(n\*fact(n-1);**

**}**

**Евклид алгоритмі бойынша екі санның ең үлкен ортақ бөлгіші табу**

**алгоритмі:**

**#include**

**#include**

**double euob(double n, double m);**

**int main ()**

**{**

**double f;**

**double n=1470;**

**double m=693;**

**f=euob(n,m);**

**printf("f=%10.0f \n”, f);return(0);**

**}**

**double euob(double n, double m)**

**{**

**w=floor(fmod(n,m);**

**if (w==0) return(m);**

**else return(euob(m,w));**

**}**