**Лекция № 5.** Рекурсиялық функциялар. Эрбран-Гёдель рекурсиялық функциялардың формалды жүйелері. Черч тезисі. Лямбда-есептеу.

**МАҚСАТЫ:** Рекурсиялық функциялар, примитивті-рекурсиялық функциялар мен предикаттар, жартылай- рекурсиялық функциялар, жалпырекурсиялық функциялар, Эрбран-Гёдель рекурсиялық функциялардың формалды жүйелері, Черч тезисі, рекурсивті функциялардың комбинаторлық анықталуы ұғымдарымен таныстыру.

**СҰРАҚТАР**:

1. Рекурсиялық функциялар
2. Примитивті-рекурсиялық функциялар мен предикаттар
3. Жартылай- рекурсиялық функциялар
4. Жалпырекурсиялық функциялар
5. Эрбран-Гёдель рекурсиялық функциялардың формалды жүйелері
6. Черч тезисі.
7. Рекурсивті функциялардың комбинаторлық анықталуы.

**НЕГІЗГІ ҰҒЫМДАР:** Рекурсивті функциялар, жартылай функция, жартылай, рекурсивті функция , қарапайым, жартылай рекурсивті функция . рекурсивті функциялардың комбинаторлық анықталуы.

Алгоритм ұғымын формальдау үшін алғаш шыққан әдіс − реккурентті теңдеулер Кең таралған. Курт Гендель енгізген. Рекуррентті теңдеулердің үлкен класының жалпы шешімі алгоритмдер теориясында қолданылуы кез-келген алфавиттегі сөздерді натурал сандардың тізбегімен нөмірлеуге негізделген. Сондықтан кез-келген алгоритмді бүтін санды аргументі бар бүтін санды функцияның мәнін есептеуге келтіруге болады. Рекуррентті теңдеулердің үлкен класының жалпы шешімі рекурсияға байланысты.

Рекурсия - функцияның берілу тәсілі, мұнда функцияның мәні берілген аргументтерде беріліп қойылған басқа аргументтердің мәнінде функция мәнімен анықталады.**Рекурсивті функциялар** үш топқа бөлінеді: қарапайым рекурсивті функциялар, жалпы рекурсивті функциялар, жартылай рекурсивті функциялар.

Мұндағы жалпы және жартылай деген ұғым функцияның анықталу облысына байланысты, яғни жартылай анықталған рекурсивті функция және барлық жерде анықталған рекурсивті функция деген мағынаны білдіреді.*(partial recursive function* - «рекурсивные функции, определенные на части множества возможных аргументов» , *total recursive function* - «рекурсивные функции, определенные на всём множестве возможных аргументов»).

Негізгі ұғымдары:

X және Y екі жиын бар. X-тің Y-ке жекелей (жартылай, частичная) функциясы деп немесе бейнеленуі деп мына жұпты айтады: <D(f),f> Мұндағы D(f) -Х-тің ішкі жиыны, оны f –тың анықталу облысы деп атайды.f: D(f) ->Y бейнелеуі деп атайды.

**Жартылай функция.** Егер D(f) бос болса, онда f еш жерде анықталмайды. Еш жерде анықталмаған жартылай функция бар деп санаймыз.N натурал сандар жиыны берілсін. (N) n мұндағы n>1, N өз-өзіне декарттық көбейтіндісін белгілейміз. Нәтижесінде мынандый жиын аламыз: xi∈N (x1, x2,... xn)

Жартылай функцияны осы (N)m ->(N)n әртүрлі m мен n үшін құраймыз. (N)m ->(N)n бейнеленетін жартылай функция f есептелетін функция деп аталады, егер оған xi∈(N)m кірістік жиынтық үшін, шығысында f(x) беретін алгоритмді (программаны) көрсете алсақ. Мұндағы х D(f) –ке тиісілі xi∈ D(f) , бұл жағдайда ол f(x) функциясын береді. Егер х∈ D(f) емес болса, онда х кірістік деректер үшін нәтижесінде 0 аламыз.

Осы анықтама бойынша алгоритмнің (программаның) формальды емес түсінігі есептелінетін функция ұғымымен байланысты. Олай болса, алгоритмнің орнына есептелінетін функция қасиеттерін зерттеуге болады.

Есептелетін функцияның орнына функцияның кең класын қолдану қажет болады, яғни жартылай есептелетін функция.

 **жартылай есептелетін функция** деп аталады, мынадай алгоритмді көрсете алсақ:

Кірістік  жартылай жиынтығы үшін шығысында f(x),  немесе алгоритм белгісіз уақытқа ұзақ жұмыс істейді, егер .

егер f функциясы есептелмейтін де, жартылай есептелмейтін де болса, онда ол **есептелмейтін жартылай функция** деп аталады.

Осы ұғымның ішінде негізгісі жартылай есептелушілік. Өйткені барлық есептелетін функция жартылай есептелуге тиесілі, ал барлық анықталатын жартылай есептелетін функция есептелінеді. (Вычислимые функции полувычислимые а всюду определенные полувычислимые функции вычислимы.)

**Жартылай рекурсивті функция** Тъюринг машинасымен есептелінетін жиындармен бірдей. Жартылай рекурсивті функция ⊃ жалпы рекурсивті функцияны, жалпы рекурсивті функция ⊃ қарапайым рекурсивті функцияны өзінде қосады. Жартылай рекурсивті функцияны кейде кәдімгі рекурсивті функцияны деп атайды.

**Қарапайым (примитивті) рекурсивті функция.** Оны анықтау үшін қарапайым рекурсивті функцияның базалық класын және екі операторды(орнына қою және қарапайым рекурсия) көрсетуден тұрады. Бұл операторлар бұрынғы бардың негізінде жаңа қарапайым рекурсивті функцияны құруға береді.

Қарапайым рекурсивті функцияның базалық түрлері:

*Нольдік функция* – аргументі жоқ функция, әрқашан нольді қайтарады.

*Ілесу функциясы* (следование) – S, кез келген натурал х санын сәйкес келетін бір айнымалыдан тұрады, нәтижесі сол саннан кейінгі сан (x+1).

; ;

n – а нымалыдан тұратын функция кез келген  натурал санның реттелген тізбегіне осы жиынның ішіндегі  санын сәйкес келтіретін функция.

**Жартылай рекурсивті функция** екі оператормен (орнына қою және қарапайым), оған қоса аргументті минимизациялау операторын қосу арқылы анықталады. f N натурал айнымалының функциясы болсын, онда f функциясының аргументінің минимумының операциясының нәтижесі айнымалыдан тұратын h функциясы:

 егер

 орындалса.

h функция f функцияның соңғы аргументінің минималды мәнін береді. Ол мәнде f=0 болады.

 III. **Жалпы рекурсивті функция.** Аргументтердің барлық мәндерін анықтайтын жартылай рекурсивті функцияның ішкі жиыны.

*Орнына қою операторы.* Бұл қарапайым рекурсивті функция және жартылай РФ қолданылады. f m айнымалының функциясы болсын, ал g1…gm әрқайсысы n айнымалысының реттелген функциясы. Онда f функциясының орнына қойылған gk функциясының нәтижесі:n айнымалының h функциясы болады. Ол x1...xn натурал сандар жиынына осы жиынтыққа жататын һ(x1...xn )=f(g1 (x1...xn)… gm (x1...xn)) сәйкес келтіреді.

*Қарапайым рекурсия операторы.* f n айнымалының функциясы, g n+2 айнымалысының функциясы, онда f және g функцияларының жұбына орындалатын қарапайым рекурсия операциялардың нәтижесі: n+1 айнымалысының h функциясы.

һ(x1...xn , 0 )= f(x1...xn);

һ(x1...xn , y+1 )= g(x1...xn , y, һ(x1...xn , y ))

Мысалдар. Қарапайым рекурсивті функцияларға жататын арифметикалық функциялар:

Екі натурал санның қосындысы sum(a,b)=a+b

Екі натурал санның көбейтіндісі mul(a,b)=a\*b

Программалау тілі мен РФ байланысы. Қарапайым РФ программадағы функцияларға сәйкес келеді, тек арифметикалық операторлар қолданып және шартты оператор мен арифметикалық цикл операторын қолданады.

Егер while операторын қолдансақ, итерация саны алдын-ала белгілі, онда жартылай РФ болады. Жартылай рекурсивті функцияның нәтижесі: тек сан емес, программалық істен шығу немесе белгісіз мәнге сәйкес келеді.

Черчтің рекурсивті функцияға байланысты тезисі

f функциясы, егер ол жартылай рекурсивті болса ғана жартылай есептеледі.

f функциясы есептелінеді сонда тек сонда, егер f рекурсивті болса және оның сипаттамалық(характер) функциясы болса. XD(f)  характерная функцияның бейнелеуі, x-тің y-тегі (x Є y) характ функция дегеніміз Черч тезисі орындалатын функция.

**Ля́мбда-есептеу** (*λ-**есептеу* ) — [формалды](http://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0%22%20%5Co%20%22%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F%20%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) жүйе, американдық математик  Алонзо Черч енгізді, есептелу ұғымын [формалдау және талдау үшін.](http://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F)

λ-есептеуді может [прототипті программалау тілдері тобы ретінде](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D1%82%D0%B8%D0%BF%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) қарастыруға болады.. Басты ерекшелігі олар жоғарлы деңгейлі тілдерге жатады. Сол арқылы аргументері мен мәндері операторлар болып табылатын операторларды зерттеуге мүмкіндік болды. Бұл топтың тілдері функционалды., функция немесе оператор ұғымына негізделген. Негізгі операциялары функциональды аппликация және и функциональды абстракция.

λ- есептеуді [Джон Маккарти](http://ru.wikipedia.org/wiki/%D0%94%D0%B6%D0%BE%D0%BD_%D0%9C%D0%B0%D0%BA%D0%BA%D0%B0%D1%80%D1%82%D0%B8) [Лисп](http://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%81%D0%BF) тілінде жүзеге асырды, Лисп-технологияда (аппараттық түрде [Лисп-машина](http://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%81%D0%BF-%D0%BC%D0%B0%D1%88%D0%B8%D0%BD%D0%B0)да) нақты жүзеге асты.

**Рекурсивті функцияның комбинаторлық анықталуы**

f(x)=x фун­кциясының комбинарторлық анықталуын көрсетейік.

*Шешімі*.

Фун­кция f(x ) m-опе­ра­то­р арқылы фун­кциядан  h(x,y)=x-y алынды. Ол тер­ммен H анықталады. f фун­кциясын анықтайық, G термін алайық, мына теңдеуді қанағаттандырады Gk m=if Zero(Hk m)then m else Gk(S+m).

Ендеше f  мына терммен F=[x].Gx0 анықталған. Тексерейік.  !f(k) болсын, онда f(k)=k. h(k,i) анықтамасы бойынша 0£i£k үшін !h(k,i)=li, мұндағы li¹0, i<k и lk=0 болсын. Gn s=s+k, болғандықтан  k , s и n – кез келген ну­ме­ра­лдар. Ендеше  Fk=([x].Gx0)k=Gk 0=0+k=k=f(k).

**ҚОРЫТЫНДЫ:**

Алгоритм ұғымын формальдау үшін алғаш шыққан әдіс − реккурентті теңдеулер Кең таралған. Курт Гендель енгізген. Рекуррентті теңдеулердің үлкен класының жалпы шешімі алгоритмдер теориясында қолданылуы кез-келген алфавиттегі сөздерді натурал сандардың тізбегімен нөмірлеуге негізделген. Сондықтан кез-келген алгоритмді бүтін санды аргументі бар бүтін санды функцияның мәнін есептеуге келтіруге болады. Рекуррентті теңдеулердің үлкен класының жалпы шешімі рекурсияға байланысты.