**Парамонов Павел.**

Лабораторная работа №1

**Задание 1.**

1. λxy.xz

Проведем β-редукцию со значением “at”

λxy.xz -> (λx.λy.xz)at -> [x:=a]-> (λy.az)t->az

1. λxz.xz -> (λx.λz.xz)at ->[x:=a]-> (λz.az)t->at => тк результат β-редукции не совпадает, не альфа эквивалентное
2. λmn.mz -> (λm.λn.mz)at ->[m:=a]-> (λn.az)t->az => тк результат β-редукции совпал, данное лямбда-выражение - альфа эквивалентное
3. λz(λx).xz -> (λz.λx.xz)at -> [z:=a]->(λx.xa)t->[x:=t]->ta => => тк результат β-редукции не совпадает, не альфа эквивалентное.
4. λxy.xxy

Проведем β-редукцию со значением “ft”

(λx.λy.xxy)ft->(λy.ffy)t->fft

1. λmn.mnp->(λm.λn.mnp)ft ->(λn.fnp)t->ftp => не альфа эквивалентное
2. λx(λy).xy-> (λx.λy.xy)ft->(λy.fy)t->ft => не альфа эквивалентное
3. λa(λb).aab -> (λa.λb.aab)ft->(λb.ffb)t->fft=> альфа эквивалентное
4. λxyz.zx

Проведем β-редукцию со значением “abc”

(λx.λy.λz.zx)abc-> (λy.λz.za)bc-> (λz.za)c->ca

1. λx.(λy).(λz) => не альфа эквивалентное
2. λtos.st->(λt.λo.λs.st)abc->(λo.λs.sa)bc->(λs.sa)c->ca=> альфа эквивалентное
3. λmnp.mn-> (λm.λn.λp.mn)abc->(λn.λp.an)bc->(λp.ab)c->ab => не альфа эквивалентное

Ответ: 1 - b; 2 – c; 3 – b.

**Задание 2**

1. λx.xxx – является комбинаторным выражением, т.к отсутствуют свободные переменные, а есть только связанные;
2. λxy.zx – не комбинаторное, тк z – свободная переменная;
3. λxyz.xy(zx) - является комбинаторным выражением, т.к отсутствуют свободные переменные;
4. λxyz.xy(zxy) - является комбинаторным выражением, т.к отсутствуют свободные переменные;
5. λxy.xy(zxy) - не комбинаторное, тк z – свободная переменная;

**Задание 3**

1. λx.xxx – уже в β-НФ, т.к отсутствуют аргументы, к которым можно применить абстракцию;
2. (λz.zz)(λy.yy) -> [z:= λy.yy]->(λy.yy)(λy.yy)->[y:= λy.yy]->(λy.yy)(λy.yy)…

выражение расходится, т.к не имеет остановы.

1. (λx.xxx)z -> [x:=z]-> zzz – выражение приведено к β-НФ, т.к не осталось головы

**Задание 4**

1. (λabc.cba)zz(λwv.w) -> (λa.λb.λc.cba)zz(λw.λv.w) ->[a:=z]-> (λb.λc.cbz)z(λw.λv.w)-> [b:=z] -> (λc.czz)(λw.λv.w)-> [c:= (λw.λv.w)] -> (λw.λv.w)zz->[w:=z]-> (λv.z)z-> [v:=z] -> z;
2. (λx.λy.xyy)(λa.a)b -> [x:= (λa.a)] -> (λy.(λa.a)yy)b-> [y:=b]-> (λa.a)bb->[a:=b] -> bb;
3. (λy.y)(λx.xx)(λz.zq) -> [y:= (λx.xx)]-> (λx.xx)(λz.zq)->[x:= (λz.zq)]->

(λz.zq)(λz.zq)->[z:= (λz.zq)]-> (λz.zq)q->[z:=q]-> qq;

1. (λz.z)(λz.zz)(λz.zy) ⬄(λn.n)(λm.mm)(λz.zy) -> [n := (λm.mm)]-> (λm.mm)(λz.zy) -> [m:= (λz.zy)] -> (λz.zy) (λz.zy)->[z:= (λz.zy)]-> (λz.zy)y -> [z:=y]->yy;
2. (λx.λy.xyy)(λy.y)y ⬄(λx.λy.xyy)(λm.m)t -> [x:= (λm.m)]-> (λy.(λm.m)yy)t -> [y:=t] -> (λm.m)tt -> [m:=t] -> tt;
3. (λa.aa)(λb.ba)c ⬄(λm.mm)(λb.ba)c -> [m:= (λb.ba)]-> (λb.ba) (λb.ba)c-> [b:= (λb.ba)] -> (λb.ba)ac -> [b:=a] -> aac;
4. (λxyz.xz(yz))(λx.z)(λx.a) ->(λx.λy.λz.xz(yz))(λx.z)(λx.a) -> [x:= (λx.z)] -> (λy.λz.(λx.z)z(yz))(λx.a) -> [y:= (λx.a)] -> λz.(λx.z)z((λx.a)z) ⬄ λz.(λx.m)z((λx.a)z) -> [x:=z] -> λz.(λx.m)za -> [x:=z] -> λz.ma;