Monther Monther General Collection  Theory-Livel Space nursery  LLS gen 2 = retirement  General Contents live young "  Jo SHC Often, do Mrs spaningly  Theory-generational Pointer Table program & 0, f = x  S. S.S.S. M. S.S.S. M. NOW: S.F. X. is new  fight yete off o is old  verile ight del (0, x)  burrier o.f = x  G. b. C. mem is as M4S + IAPT  there is as SEC to white barriers  Radio active decay model	2-2) Long-lived things are copied a lot
FS TS  FS	Marin - Marin
"most objects die young"  Jo Ste often, do Mts sparingly  Inter-generational Pointer Table Bogram & o, f = x  ssss Mssss M  syst iget for oild  write igpt.add(o,x)  barrier o, f = x  C. b. C: mem is as Mts & ILPT  the is as Ste + write barriers	FS TS  FS  FS  FS  Pen 0  Inursery  Norsery
"most objects die young"  Jo Ste often, do Mts springly  Inter-generational Pointer Table Program & O.f = X  ssss Mssss M  igpt ippte  write igpt.add(o,x)  barrier o.f = x  G. G. C. mem is as Mas & IGPT  time is as Ste to write barriers	gen 1 = retirement
Inter-generational Pointer Table Program & o, f = x  SSSS MSSSS M  NOW: (if x is new  ight ight ight ight ight osold  write ight, add (o, x)  barrier o, f = x  G. G. C: mem is as MAS & IGPT  time is as S+C + write barriers	
SSSS MSSSS M  igpt igpt  write	
SSSS MSSSS M  igpt to the sold of the sold	Inter-generational Pointer Table Pogram & O, f = x
ight ight write fight, add (0, x)  barrier 0, f=x  G. b. C: mem is as MAS & ILAPT  time is as SAC + write barriers	ssss Mssss M now: cif x is new
GibiC: mem is as MAS & IGPT  time is as S+C + write boarriers	igot 2 igote orsold
G.b.C: mem is as MAS & ILAPT  time is as S+C + unite barriers	
time is as StC + write barriers	
Radioactive decay model	time is as StC t write boarriers
	Radioactive decay model

```
int main (intarge, char ## argv) &
                                              = Stack Overflow
        return main (argc, argv);
    \Omega = ((\lambda x, (x \times 1))(\lambda x, (x \times 1))) \qquad \omega = \lambda x, x \times
       2 -78 52
    LD, Ø, m+7
Teck < W, Ø, < ar, w, Ø, m+>>
                                          c = <w, 0>
    < w, p, < fn, < w, 07, m+>>
    <(xx), [x\mapsto c], m+>
    < x , [x+7], < ar , x , [x+7], m+>>
                                                  constant
    < c, [x +> c], < an, x, [x +> c], m +>>
                                                   space
    < x, [x+7c], < fn, x, <w, 9>, m+7>
   ۲ د ، ۱ ۱
   < (x x), [x+7c], m+7
   SECD machine
                                                       3 4 t
                                                       7
                                       Stack-Machine
   S-stack (notall like K)
                                                      5 34 + X
   E-env (same as CEK)
                                                       35
   C - control string (very similar to CEK'S C)
   D- dump (saved SECD)
 CS, F, (MN) C, D> H> < S, E, MN ap C, D>
 (S, E, (JX,M)(,D) H) < <(JX,M),E) S, E, C, D>
 \langle S, E, b c, D \rangle \longrightarrow \langle b S, E, C, D \rangle
 \langle S, E, X C, D \rangle \longrightarrow \langle E(X) S, E, C, D \rangle
 < V < \X,M, Eclo> S, E, ap C, D>
        HT CE, ECIO[XHV], M, <S, E, C, D>>
 < V S , E , E , < S', E', C', D' > 7
      → < V S', E', C', D'>
```

15-2/ 12 = (w w)  $w = (\lambda x, x x)$ bytecode = bytecode C= < w, Ø> (E, O, D, m+7 [ ] = [w] Lw] ap H> KE, Ø, wwap, m+>  $Lw = (-1x, x \times ap)$ H> < < W, Ø7, Ø, wap, m+7 = < c, ø, wap, m+> vseless (dx, xxap) (-lx, xxap) ap < C C , Ø , ap , m+> () H> < E, [XH>C], (XX), < E, Ø, 2, m+>> e, [xmc], xxap, A> HOCC, [XHO], X ap, A) HOCCC, [XHC], ap, FT> H) ( E, [XH), (XX), (E, [XH), E, #>> useless CEK allocates on any evaluation, frees on fun calls SECD allocs on calls and arg eval "O(a) 0(a+c) Trist (FF) 0) Vntrust == (Af, (Ax. (((++) (1x.x))) CTRUST, CUNTRUST, ... (TRUST?) < (MN), E, K) FOR < M, E, Car, N, E, K>> < (JXIM), E, K) >> < < (JXIM), E>, E, K> Els = E example S (restricting E to S) safe Exm1, ym23/2x3 = Exm13 for Space " OH < M, E| FU(m), < an, N, E| FV(N), K>> (2) H) < < (1X, m), El FV(m) 7, Ø, K) prevents sharing (c.f. "flat closures")

