

```

0 ( overlay prefix-----DAMAGE-----)
1 \ basic routines
2 \ instance and species variables: VESSEL
3 \ instance and species variables: SHIP
4 \ crew variables
5 \ dist
6 \ refresh auxilliary screen
7 \ calculate damage from alien laser
8 \ damage a crew member- setup
9 \ wounded array
10 \ end game and hull check
11 \ random# to damage field case
12 \ random# to damage field case
13 \ find-resource
14 \ .wounded
15 \ crew-damage
16 \ crew-damage
17 \ pdamage
18 \ do-pdamage, do-pldamage
19 \ mcalc drain-shield
20 \ drain-armor
21 \ hull-damage, etal
22 \ random# to damage area case
23 \ distribute damage mechanism
24 \ do-damage for a particular missile, to ship
25 \ explode, for missiles
26 \ do-damage for a particular missile, to ship
27
28 \ which missile damage routine to execute
29 \ which laser damage routine to execute
30
31

```

```

0 ( overlay prefix-----DAMAGE-----((rfg18oct85) \ instance and species variables: SHIP (rfg22oct85)
1 20 c: sship
2 5 width ! sship 17 1 ifield: %hull-pts sship 18 1 ifield: %arm-cla
3 vocabulary damage-voc immediate sship 19 1 ifield: %engine-pts sship 20 1 ifield: %eng-cla
4 133 open-overlay sship 21 1 ifield: %sensor-pts sship 22 1 ifield: %sen-cla
5 damage-voc definitions sship 23 1 ifield: %comm-pts sship 24 1 ifield: %com-cla
6 1500 trans-allot newt-dp sship 25 1 ifield: %shield-pts sship 26 1 ifield: %shd-cla
7 sship 27 1 ifield: %missile-pts sship 28 1 ifield: %mis-cla
8 sship 29 1 ifield: %laser-pts sship 30 1 ifield: %las-cla
9 sship 48 2 ifield: shield-cnt sship 69 4 ifield: %status
10 sship 50 2 ifield: armor-cnt
11 sship 73 1 ifield: %heading
12
13 61 20 1 ifield: %ships-destroyed
14
15

```

```

1 4
0 \ basic routines (rfg02oct85) \ crew variables (rfg18oct85)
1 head: >flag t: 0= 0= t; \ insures Obit set 16 31 1 ifield: ^vit \ crewmember vitality
2 16 19 1 afield: ^dur
3 head: rand% t: 1 100 rrnd t; 16 11 15 ifield: ^name
4
5 head: ship-coords \ --- x,y create rsrc 9 allot \ holds resource#s
6 t: xabs @ yabs @ t; v: #rsrc #rsrc off \ # of resources damaged this time
7
8 head: ves-coords \ --- x,y ! assume pointed to ci vessel icon v= mclass \ missile class: ditto below
9 t: @ix @iy t; 2v= mcrds \ gets stuffed when missile call is made by combat
10
11 head: @xy t: inst-x @ inst-y @ t; head: @mclass t: mclass c@ 15 and t;
12 head: @rclass \ resource addr --- value
13 head: ship>c t: #ship >c+s t; t: c@ 7 and t;
14
15

```

```

2 5
0 \ instance and species variables: VESSEL (rfg22oct85) \ dist (rfg22oct85)
1 decimal 25 c: vessel head: mcoords t: mcrds d@ t;
2 \ returns missile coords, as stuffed
3 vessel 12 2 afield: ves-armorpts
4 vessel 14 2 afield: ves-shieldpts head: dist \ x,y,x1,y1 --- distant
5 vessel 18 1 afield: ves-missile t: rot - dup * >r
6 vessel 19 1 afield: ves-laser - dup * 0 r> 0 d+ sqrt t;
7 vessel 21 1 afield: ves-element
8 head: 0max-! \ decrement,field --- sets field to 0 or more
9 vessel 12 1 ifield: ves-speed t: >r r@ ifld@ swap - 0 max r> ifld! t;
10 vessel 23 4 ifield: ves-last-fired
11 vessel 18 2 ifield: ves-armor-hits head: +!destroy
12 vessel 20 2 ifield: ves-shield-hits t: (encounter) 1.5@ >c+s
13 vessel 11 1 ifield: ves-heading 1 #ships-destroyed dup c@ rot + swap c! iclose t;
14 vessel 18 1 ifield: ves-debris
15

```

6

9

```

0 \ refresh auxilliary screen          (rfq13oct85). \ wounded array          (rfq30oct85)
1                                     create wounded 6 allot \ hold offset into roster of injured
2 ' ovdbars c: #ov/damage
3 ' ov/status c: #ov/status
4 ' nop c: #nop
5
6 case #aux>module
7 1 is #ov/status 3 is #ov/damage others #nop
8
9 head: refresh-aux t: #aux @ #aux>module module t;
10
11 head: ?shields-up
12 t: %status c@ 8 and >flag t;
13
14
15

```

7

10

```

0 \ calculate damage from alien laser  (rfq18oct85) \ end game and hull check          (rfq22oct85)
1
2 create ldamage 0 , 30 , 70 , 120 , 170 , 220 ,
3 \ damage per laser class
4
5 head: lcalc \ parameter --- damage points ( vessel --- vessel ) t: yellow !color bfill red !color
6 t: >r @xy ship-coords dist          " HULL DESTROYED - RESTART GAME " .tty
7 70 swap -
8 r> ifld@ 2* ldamage + @ *
9 70 / t;
10
11 head: alien-lcalc t: ' ves-laser lcalc t;
12
13 head: player-lcalc t: ' %las-cls lcalc t;
14
15

```

8

11

```

0 \ damage a crew member- setup        (rfq18oct85) \ random# to damage field case          (rfq18oct85)
1 v: #crew-damage
2
3 head: ?injure \ -- flag! true if are to injure
4 t: rand% 70 < t;
5
6 head: #injure \ --- damage
7 t: 10 #crew-damage @ 1+ rrnd
8 ^dur c@ / t;
9
10
11
12
13
14
15

```

12

```

0 \ random# to damage field case (rfq09oct85)
1 case #>?field \ check field for existance of resource
2 1 is hull-check
3 2 is engine-check
4 3 is sensor-check
5 4 is comm-check
6 5 is miss-check
7 6 is laser-check
8 7 is crew-check
9 8 is shield-check
10 others 0= \ return false
11 \ returns true flag if resource exists (to be destroyed!)
12
13
14
15

```

13

```

0 \ find-resource (rfq18oct85)
1 head: ?any-resources \ --- flag! true if any on player's ship
2 t: rsrc 9 0 fill #rsrc off
3 0 9 1 do i #>?field dup
4     if i #rsrc @ rsrc + c! 1 #rsrc +!
5     then or loop t; \ also sets lookup array for choosing
6
7 head: find-resource
8 t: ?any-resources
9     if 0 #rsrc @ rrnd rsrc + c@ \ holds valid resource#
10     then t;
11
12
13
14
15

```

14

```

0 \ .wounded (rfq09oct85)
1 head: wounds \ position set by .tty of name
2 t: ^vit c@ dup >r 0=
3     if " KILLED"
4     else r@ 1 2! within
5         if " CRITICALLY "
6         else r@ 2! 5! within
7             if " HEAVILY "
8             else r@ 5! 8! within
9                 if " MODERATELY "
10                 else " SLIGHTLY "
11                 then
12                     then
13                     then type " WOUNDED "
14                     then type r> drop t;
15

```

15

```

\ crew-damage (rfq22oct85)
head: .wounded
t: #wounded @ ?dup
    if white !color
        0 do i wounded + c@ 3* roster + 1.5@ 2dup or
        if >c+s ^name count -trailing .tty
            ." IS " wounds iclose
        else 2drop
        then loop
    then #wounded off t;

: destroy-ves 43 !id 8 ves-heading c!
white !color >display " ALIEN SHIP DESTROYED " .tty
0 ves-speed c! \ lose damage: now debris !
ves-element ves-debris 7 move 24 !inst-species +!destroy ;

```

16

```

\ crew-damage (rfq18oct85)
head: crew-damage \ damage --- \ apply to crew members
t: #crew-damage ! init-wound
    roster dup 18 + swap
    do i 1.5@ 2dup or
        if ?injure
            if >c+s #injure
                ^vit 0max-! iclose
                i roster - 3 / !wound
            else 2drop
            then
        else 2drop leave ( never any gaps in roster )
        then 3 +loop t;

```

17

```

\ pdamage (rfq22oct85)
: pdamage \ damage --- ( vessel --- vessel )
    ves-shield-hits @
    ves-armor-hits @ null d= \ get both fields!!
    if drop destroy-ves \ no more damage possible
    else ves-shield-hits @ 0=
        if ^ves-armor-hits 0max-!
        else ^ves-shield-hits 0max-!
        then
    then ;

```

18

```

0 \ do-pdamage, do-pldamage
1
2 head: do-pdamage \ coords,damage --- ! apply to alien vessels
3 t: >r ilocal @ ?icons-at ?dup if
4   r@ swap 0 do beep
5   dup rot point)icon
6   @il @ih >c+s
7   @inst-species 24 = not \ avoid dead ships
8   if pdamage
9   else drop \ no damage to our own ship
10  then iclose loop drop then r> drop t;
11
12 head: do-pldamage t: player-lcalc do-pdamage t;
13
14
15

```

(rfg22oct85) \ hull-damage, etal

21

(rfg18oct85)

```

head: hull-damage \ damage --- ! affect hull
t: ' %hull-pts 0max-! t;

head: engine-damage t: ' %engine-pts 0max-! t;
head: sensor-damage t: ' %sensor-pts 0max-! t;
head: comm-damage t: ' %comm-pts 0max-! t;
head: miss-damage t: ' %missile-pts 0max-! t;
head: shield-damage t: ' %shield-pts 0max-! t;
head: laser-damage t: ' %laser-pts 0max-! t;

\ shields already taken care of in draining

```

19

```

0 \ mcalc drain-shield
1 create Mdamage 0 ,
2 200 , 400 , 700 , 1000 , 1500 , 4000 ,
3 \ plasma bolt acts like a class 6 missile
4
5 head: mcalc \ damage done by missile
6 t: @mclass 2* mdamage + @ t;
7
8
9 head: drain-shield \ damage --- ! drain the player's shield
10 t: shield-cnt @ swap
11 6 %shd-clc @rclass - * 5 / \ compute actual drainage
12 - 0 max shield-cnt ! t;
13
14
15

```

(rfg22oct85) \ random# to damage area case

22

(rfg07oct85)

```

case #>damage
1 is hull-damage
2 is engine-damage
3 is sensor-damage
4 is comm-damage
5 is miss-damage
6 is laser-damage
7 is crew-damage
8 is shield-damage
others drop

\ all the routines consume damage amount

```

20

```

0 \ drain-armor
1
2 head: drain-armor \ damage --- ! drain the player's armorshield
3 t: armor-cnt @ swap
4 6 %arm-clc @rclass - * 5 / \ compute actual drainage
5 - 0 max armor-cnt !
6 armor-cnt @ 0=
7 if ' ov/damage module 3 #aux ! then t;
8
9
10
11
12
13
14
15

```

(rfg18oct85) \ distribute damage mechanism

23

(rfg30oct85)

```

head: #damage \ initial damage --- remainder,actual damage
t: dup 1 swap 1+ rrnd 10 max
  dup >r - 0 max r> t;

: (do-damage)
  shield-cnt @ >flag ?shields-up and
  if drain-shield
  else armor-cnt @
    if drain-armor
    else 10 %hull-pts c@ * / \ reduced by 10
      3 #aux ! find-resource #>damage \ force damage display
    then
  then ;

```


24

```

0 \ do-damage for a particular missile, to ship      (rfq30oct85)
1 head: ?refresh \ update display only if shields depleted
2 t: shield-cnt @ armor-cnt @ null d= drop 1 ( always, for now)
3 if refresh-aux then t;
4
5 : do-adamage \ damage --- ! distributed among resources
6 begin ?dup 0= not
7 while #damage (do-damage)
8 repeat #wounded @
9     if .wounded ' (obits) module ' (role) module
10    then ?refresh
11 ' %hull-pts ifid@ 0= if end-game then ;
12
13 head: do-aldamage
14 t: alien-lcalc ship>c do-adamage iclose t;
15

```

27

25

```

0 \ explode, for missiles      (rfq22oct85) \ which missile damage routine to execute      (rfq18oct85)
1
2 head: !parms t: mclass c! mcrds d! t;
3
4 head: explode \ color ---
5 t: >mainview !color
6 mcoords wld>scr scr>blt
7 swap 4 + swap
8 xwld:xpix @ 6 min 2 ( 1 ) do 2dup i .circle 2 +loop 2drop \ expects x,y,class,origin
9 v>display
10 ( >0font >alpha quit) t;
11
12
13
14
15

```

28

26

```

0 \ do-mdamage for a particular missile, to ship      (rfq22oct85) \ which laser damage routine to execute      (rfq18oct85)
1
2 head: damage-disp \ color ---
3 t: explode white !color
4 >display @mclass dup 6 =
5 if drop " PLASMA BOLT" .tty
6 else " MISSILE CLASS " .tty . then t;
7
8 head: do-pmdamage
9 t: !parms ship>c lt-blue explode
10 mcoords mcalc do-pdamage iclose ( black explode ) t;
11
12 head: do-amdamage
13 t: !parms red damage-disp
14 mcalc ship>c do-adamage iclose ( black explode ) t;
15

```

29

30

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

31

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

32

0 (overlay suffix-----DAMAGE----- (rfg02oct85)
1
2 dispose
3 close-overlay
4 133 overlay damage-ov damage-ov
5
6 forth definitions
7 : do-damage damage-ov damage-voc do-damage save-overlay ;
8
9 ov-cancel
10 : dm ov-cancel damage-ov ;
11
12
13
14
15