```
(RPAQQ TYPEHAXCOMS ((FUNCTIONS ALLOCATE-11-BIT-TYPES ALLOCATE-SUPER-CHAIN ALLOCATE-TO-TYPE-NUMBER
                               ALLOCATE-WITH-NAME COLLECT-SUPER-CHAIN TEST-TYPEP)))

(DEFUN ALLOCATE-11-BIT-TYPES NIL (ALLOCATE-TO-TYPENUMBER 1023)
```

;;; allocates typenumber 1023, then allocates a type named "realbig", and checks it's instances

```
        (ALLOCATE-WITH-NAME (QUOTE REALBIG))
        (CL:SETQ AREALBIG (NCREATE (QUOTE REALBIG)))
        (TYPENAMEP AREALBIG (QUOTE REALBIG))
        (EQ (NTYPX AREALBIG)
            1024))

(DEFUN ALLOCATE-SUPER-CHAIN (DEPTH &OPTIONAL (SUPER* (QUOTE SUPER*-TYPE))
                                             (ROOT (QUOTE ROOT-TYPE)))
```

;;; Allocates datatypes up to datatype x inclusive.

```
        (LET [(SUPER (CAAR (DECLAREDATATYPE SUPER* (QUOTE (POINTER))
                                 NIL NIL NIL]
            [DOTIMES (I (- DEPTH 1))
                    (SETQ SUPER (CAAR (DECLAREDATATYPE (GENSYM (QUOTE TEST))
                                           (QUOTE (POINTER))
                                           NIL NIL SUPER]
            (DECLAREDATATYPE ROOT (QUOTE (POINTER))
                    NIL NIL SUPER)))

[DEFUN ALLOCATE-TO-TYPE-NUMBER (X)
```

;;; Allocates datatypes up to datatype x inclusive.

```
        (LET ((REMAINING (- X \MaxTypeNumber)))
            (CL:IF (< REMAINING 1)
                   (CL:ERROR "There are already ~D datatypes." \MaxTypeNumber)
                   (PROGN (DECLAREDATATYPE (QUOTE TEST-SUPER)
                                   (QUOTE (POINTER))
                                   NIL NIL)
```

                          ;; declare a super for the rest of the types.

```
                          (DOTIMES (I REMAINING)
                                  (DECLAREDATATYPE (GENSYM (QUOTE TEST))
                                          (QUOTE (POINTER))
                                          NIL NIL (QUOTE TEST-SUPER]

[DEFUN ALLOCATE-WITH-NAME (TYPENAME &OPTIONAL (SUPER (QUOTE TEST-SUPER)))
        (ETYPECASE TYPENAME (SYMBOL (DECLAREDATATYPE TYPENAME (QUOTE (POINTER))
                                        NIL NIL SUPER]

(DEFUN COLLECT-SUPER-CHAIN (ROOT)
        (CL:DO* ((TYPE ROOT SUPER)
                 (SUPER (GETSUPERTYPE TYPE)
                        (GETSUPERTYPE TYPE))
                 (SUPER-CHAIN NIL))
                ((NULL SUPER)
                 SUPER-CHAIN)
                (CL:PUSH SUPER SUPER-CHAIN)))

[DEFUN TEST-TYPEP (TYPE)
```

;;; ensures that instances of TYPE are instances of all its supertypes.

```
        (LET ((INSTANCE (NCREATE TYPE)))
            (CL:MAPCAR (CL:FUNCTION (CL:LAMBDA (TYPE)
                                        (TYPEP INSTANCE TYPE)))
                    (COLLECT-SUPER-CHAIN TYPE]
```