# Datatypes

(spec = specification)

speedcode\_spec:

Defines the number of repeats (z,y,x,part) and a pointer to the speedcodepart specifications.

typedef struct

{

union

{

unsigned int arr[NUMOFLOOPS];

//Unnamed structeres can't be initialized so that the arr must be declared before the structure

struct

{

unsigned int z,y,x,part;

};//Anonymous struct, feature of C99

}repeat;

//This union must be the first element of the struct speedcode, because some code relies on this

speedcodepart\_spec \*codedefines;

}speedcode\_spec;

speedcodepart\_spec:

Defines a pointer to the speedcode\_template (the small code sequence which is repeated over z,y,x, and part), the size of the speedcode\_template, a callback function for modifying the speedcodepart and a pointer to parameters for the modification function.

typedef struct

{

void\* assembly\_code;

size\_t size;

} speedcode\_template\_spec;

typedef struct

{ speedcode\_template\_spec; //Anonymous Structure, C99 feature

unsigned int repeatcount;

void (\*callback)(register void\*);

void \*parameters;

} speedcodepart\_spec;

speedcode\_template:

A small code sequence which is to be repeated many times in the generated code.

## How the speedcode\_template is to be replicated

* for z = 0 to
  + for y = …
    - for x = …
      * for part = …
        + for partrepeat = …

one speedcode part

* + - * + next
      * next
    - next
  + next
* next