

Module Braid

1. Type décrivant une tresse d'après la liste de ses générateurs dans la présentation d'Artin, et son nombre de brins (champ `size`)

```
type braid = {word : int list; size : int}
```

2. Inverse une tresse dans B_n en prenant les opposés des générateurs dans l'ordre inverse

```
let inv braid = {word = List.map (fun x → -x) (List.rev braid.word); size = braid.size}
```

3. Concaténation de deux mots de tresses

```
let concat b1 b2 = {word = b1.word @ b2.word; size = max b1.size b2.size}
```

```
let (++) = concat
```

4. Fonction utilitaire retournant le type `braid` correspondant au mot (et à la taille optionnelle) fourni

```
let make_braid ?(size = -1) w =
  let s = (if size = -1 then 1 + List.fold_left (fun a x → max a (abs x)) 0 w
           else size) in
  {word = w; size = s}
```

5. Génération de tresses aléatoires

```
let random_braid n l =
  Random.self_init ();
  let w = ref [] in
  for i = 0 to l - 1 do
    w := ((if Random.bool () then -1 else 1) ×
          (Random.int (n - 1) + 1)) :: w
  done;
  {word = !w; size = n}
```

```
let random_lower_braid n l =
  let b = random_braid (n/2 - 1) l in
  {word = b.word; size = n}
```

```

let random_upper_braid n l =
  Random.self_init ();
  let w = ref [] in
  for i = 0 to l - 1 do
    w := ((if Random.bool () then -1 else 1) ×
          (Random.int ((n/2) - 1) + (n/2) + 1)) :: !w
  done;
  {word = !w; size = n}

```