

```

    const uint64_t L = 100; // each 3 bytes. uint64_t L; uint64_t J;
    for (int i = 0; i < mc_steps; i++) { sfmt_t sfmt; // Random number generator
        behind) * L); guide_left = malloc(sizeof(int64_t));
        r = {0,1} uint64_t random_bit;
        base << 3; base += RA;
        t * J_up, uint64_t * hash;
        sh << 10; hash = (uint64_t *) malloc(sizeof(uint64_t) * L);
        guide_up = NULL;
        guide_left = NULL;
        free(guide_front);
        if (guide_left != NULL) free(guide_left);
        t = malloc(sizeof(*xg));
        t = malloc(sizeof(*xgu));
        guide_down[i] = -L; guide_behind[i] = -L-1; guide_down[0] = +L*(L-1); guide_behind[0] = NaN;
        if (beta < 0) DIE("Intended to be negative beta");
        l = exp(-4*beta); SG->probs[3] = exp(0*beta);
        ); } // Obtain a spin glass in a random state
        L = L; newnet.mc_steps = 0; set_h = 0;
        (int64_t i=0; i<L*L*L;i++){ newnet.spins[i] = rand() % 2;
        sizeof(*newnet.spins) * L*L*L);
        random_bits(); newnet.J_up[i] = rand() % 2;
        i=0; i<L*L*L;i++){ if (newnet.J_right[i] == newnet.J_left[i]) iden++;
        J_front[i] == newnet.J_right[0]) iden++;
        } void free_spin_glass(struct net* spinglass) {
            free(sp->J_right); spinglass->J_right = NULL;
            local_energy = 0;
            int64_t Ju = 0;
            for (int i = 0; i < L; i++) {
                int64_t idx, int64_t x, int64_t y;
                = SG->J_front; uint64_t i;
                left[x][y]; uint64_t up = 0;
                uint64_t front = S[idx]^S[y];
                return right + left + up + c;
                for (int64_t y=0; y<L; y++){
                    cast8_t spinidx=0; spinidx+=S[idx]*S[y];
                    s[Pidx]; if (Pidx <= 0) continue;
                    else if (Pidx > 0) continue;
                    J. } curr_idx++;
                    "n");} for (int64_t i=0; i<L*L*L;i++){
                        it64_t array, int64_t A = log10(from);
                        const int *a = a;
                        ay[0]), int_comp);
                        ){ array[i] = ar;
                            acc = 0; for (int i = 0; i < samples; i++) {
                                ccdf = 0; for (int i = 0; i < samples; i++) {
                                    double (*fur);
                                    ples to take
                                    s[samples];
                                    size+1)); r
                                    ples)); }
                                    (int64_t i=0; i<L*L*L;i++){
                                        e current
                                        L*L*L)/
                                        ; si++
                                }
                            }
                        }
                    }
                }
            }
        }
    }
    return 0;
}

```

*Trabajo de fin de grado*

# Rejuvenecimiento y memoria en vidrios de espín desde la perspectiva de JANUS

*Alejandro Clavero Álvarez*

*Director: Sergio Perez Gaviro*  
*Ponente: Alfonso Tarancón Lafita*