Problem Set 4: The BGG correspondence, Part 2

Fix $n \geqslant 0$. Let k be a field. Let $S = k[x_0, \ldots, x_n]$, and $E = \Lambda_k(e_0, \ldots, e_n)$, with gradings given by $\deg(x_i) = 1$ and $\deg(e_i) = -1$. Recall that $\omega_E \coloneqq E^* \cong E(-n-1)$.

1. (Short exercise) Prove that $\mathbf{R}(S)$ is an injective resolution of k. Conclude that

$$\mathbf{R}(S)^\vee(-n-1) \coloneqq \underline{\mathrm{Hom}}_E(\mathbf{R}(S), E)(-n-1)$$

is a free resolution of k.

Aside: the free resolution $\mathbf{R}(S)^\vee$ is an example of the *Priddy resolution* of the residue field over a Koszul algebra, which you will see in general later in Claudia's lectures.

2. Suppose $n = 1$, and let F denote the following complex of free E-modules:

$$\omega_E \xrightarrow{\begin{pmatrix} e_0 \\ e_1 \end{pmatrix}} \omega_E(-1)^2 \xrightarrow{\begin{pmatrix} 0 & e_1 \end{pmatrix}} \omega_E(-2) \xrightarrow{e_1} \omega_E(-3) \xrightarrow{e_1} \cdots .$$

Find a graded S-module M such that $\mathbf{R}(M) \cong F$. If you like, you can check your answer using `Macaulay2` (see the last problem for a demo of how to compute with $\mathbf{R}$ in `Macaulay2`).

3. Prove Eisenbud-Fløystad-Schreyer's *Reciprocity Theorem*, which is stated as follows. Let M be a graded S-module and N a graded E-module. Prove that $\mathbf{L}(N)$ is a free resolution of M if and only if $\mathbf{R}(M)$ is an injective resolution of N. Hint: $\mathbf{L}$ and $\mathbf{R}$ are exact, and so they preserve quasi-isomorphisms.

4. Let M be a finitely generated graded S-module and N a finitely generated graded E-module. Let M^* (resp. N^*) denote the graded S-module $\underline{\mathrm{Hom}}_k(M, k)$ (resp. graded E-module $\underline{\mathrm{Hom}}_k(N, k)$). The left module actions are defined in the same way as the left module action on E^* we discussed earlier (except there is no sign over S).

(a) Prove that the complexes $\mathbf{L}(N^*)$ and $\mathbf{L}(N)^\vee$ have isomorphic terms.

(b) If you're feeling ambitious, prove that there is an isomorphism $\mathbf{L}(N^*) \cong \mathbf{L}(N)^\vee$ of complexes.

Conclude from part (b) that $\mathbf{L}(\omega_E) = K(x_0, \ldots, x_n)$.

Note: one similarly shows that there is an isomorphism $\mathbf{R}(M^*) \cong \mathbf{R}(M)^\vee \coloneqq \underline{\mathrm{Hom}}_E(\mathbf{R}(M), E)$ of complexes of graded E-modules.

5. Let C be a complex of graded S-modules. The *homological shift* $C[i]$ of the complex C is the complex with $C[i]_j = C_{i+j}$ and $d_{C[i]} \coloneqq (-1)^i d_C$.

(a) Prove that, if $M \in \mathrm{Mod}(S)$, considered as a complex concentrated in homological degree 0, then $\mathbf{R}(M[i]) \cong \mathbf{R}(M)[i]$ for all $i \in \mathbb{Z}$.

(b) Prove also that $\mathbf{R}(M(i)) \cong \mathbf{R}(M)(i)[-i]$ for all $i \in \mathbb{Z}$

Note: these statements extend to complexes verbatim, but you need not prove this. One can prove in the same way that, given a complex C of graded E-modules, we have $\mathbf{L}(C[i]) \cong \mathbf{L}(C)[i]$, and $\mathbf{L}(C(i)) \cong \mathbf{L}(C)(i)[i]$. If you have time, prove these identities, and/or verify them via some examples in `Macaulay2`.

6. As a demo of how to compute with the functor $\mathbf{R}$ in `Macaulay2`, let's verify computationally that $\mathbf{R}(S)$ is an injective resolution of k.

(a) First, we need to load two packages:

```
needsPackage "BGG"
needsPackage "Complexes"
```

(b) Next, we build our polynomial ring and exterior algebra. I will work with four variables, but you can toggle this choice.

```
n = 3
Edegrees = for i from 0 to n list -1
S = ZZ/101[x_0..x_n]
E = ZZ/101[e_0..e_n, Degrees => Edegrees, SkewCommutative => true]
```

Notice that we add an optional input in the last line to make the degrees of the exterior variables -1. The default is to make the degree of each variable 1.

(c) Now we build the maps in $\mathbf{R}(S)$. The function `bgg(i, M, E)` builds the $(-i)^{\text{th}}$ differential in $\mathbf{R}(M)$. Let's make a list of the first few differentials in $\mathbf{R}(S)$.

```
L = for i from -5 to 0 list bgg(-i, S^1, E);
```

(d) Finally, we build a complex out of this list of matrices, and we compute its homology.

```
I = complex(L, Base => -6) ** E^{{-n-1}}
presentation HH_0 I
for i from -5 to -1 do print (HH_i I == 0)
```

Couple things here: the optional input `Base => -6` makes our complex live in the right homological degrees, and tensoring with $E(-n-1) \cong E^*$ makes it live in the correct internal degrees.